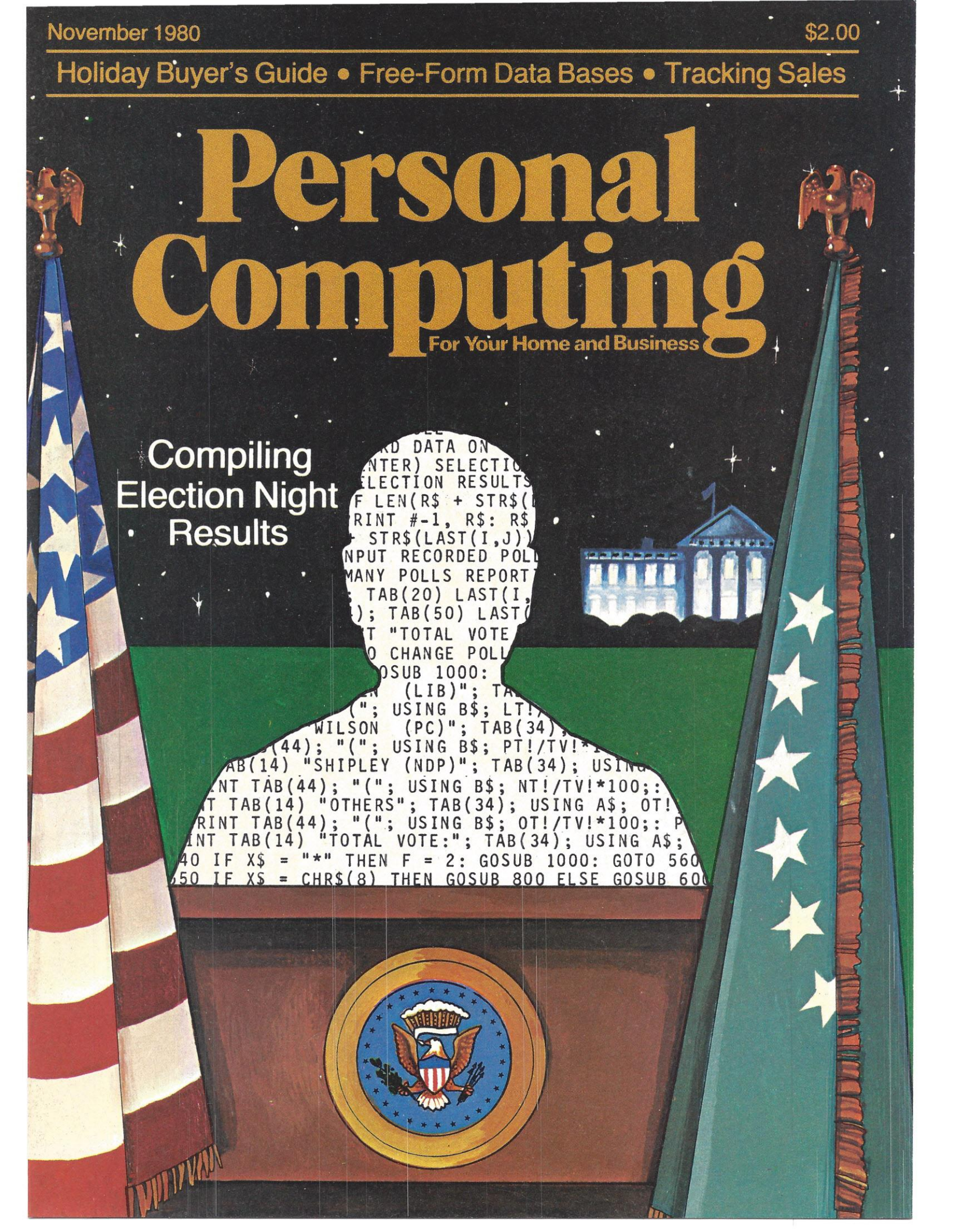
November 1980 $2.00

Holiday Buyer's Guide • Free-Form Data Bases • Tracking Sales

Personal Computing

For Your Home and Business

Compiling Election Night Results



Heath won't leave you out in the cold

You can buy a computer from just about anybody these days. And that's all you'll get. A computer. When you buy a Heath computer, you get total support. Every Heath product is backed in six ways:

1. Integrated Hardware – from a single source. No more shopping all over town – or all over the country. Whatever hardware you need, you'll find it at Heath. And when you buy from Heath, you get a balanced, harmonious computer system. Heath components are designed to work together.

2. Educational Courses – general courses that help businessmen, students and hobbyists understand what computers can do for them. Programming language courses that enable you to use the full versatility of your computer.

3. Excellent documentation – all Heath hardware and software is thoroughly documented, so no matter what your skill level, you can learn to use the product to its full potential. All documentation is clearly written and organized.

4. Software for almost every need – Choose from one of the most complete selections of software, including complete operating systems software, applications software, programming languages and professional software like word processing and accounting. Heath Users' Group offers more than 500 exciting programs.

5. Expert advice – as close as your telephone or mailbox. Technical consultants can answer your Heath hardware and software questions, by phone or by letter. Who else offers you such convenient technical advice?

6. Service support – available at 55 convenient locations nationwide. If you should ever need service, you'll find it at retail stores in the U.S., Canada and overseas, as well as at the Heath factory.

Don't be left out in the cold. Get your copy of our latest catalog and warm up to the world of Heath computers. Just write to the address below, or pick one up at the Heathkit Electronic Center* nearest you.

Heath®

Write to: Heath Company, Dept. 352-714
Benton Harbor, MI 49022

In Canada contact: Heath Company,
1480 Dundas St. E., Mississauga, Ontario L4X 2R7

*(*Heathkit Electronic Centers are located throughout the U.S. and Canada. See the white pages of your phone book for the location nearest you. In the U.S., Heathkit Electronic Centers are units of Veritechnology Electronics Corporation.)*

CP-190

Circle 2

Mountain Hardware makes more peripherals for the Apple Computer than Anybody.*

and . . .
a place to put them

INTROL X-10

Intelligent Home Controller for lights and appliances. Real-time schedules and energy conservation. Complete applications software package. Home security with random scheduler. Power usage accounting package for home energy cost control. No wiring required.

APPLE CLOCK

Real-time and date information. Interrupts permit Foreground/Background operation of two programs simultaneously. Battery back-up. Crystal-controlled for ± .001% accuracy. Onboard ROM for easy access from BASICs. Supports PASCAL. Time from one millisecond to one year.

SUPERTALKER SD200

Input/Output Speech Digitizer. Permits talking programs. I/O capability allows interactive programs with speech-prompted inputs. Use output for speech directed activities in business systems, announcements in a control-room, or sound effects in entertainment programs. Easy to use because input as well as output is under user control with special software operating system.

ROMWRITER

Program your own EPROMs. Create your own firmware. Programs 2K, 2716 5V EPROMs. Disk software package provides easy EPROM programming. EPROMs are verified after BURN. RUN your programs from on-board socket or install them on ROMPLUS+.

ROMPLUS+

More power for your system through firmware. Six sockets accept 2716 EPROMs or ROM equivalents. Six or any combination can be used at once. Scratch-pad RAM and two TTL connectors. Special 2K ROMs available for powerful system enhancement: Keyboard Filter ROM—COPYROM—Others coming soon.

MusicSystem

Sophistication previously available only on experimental mini and mainframe computer synthesizers. Digital instrumental music synthesizer system. 16 voices in stereo. Instrument definitions simulate the sound of real instruments—and more. Fully programmable waveforms. Envelope Control. Composition system—sheet music input using standard music notation. Chords and multi-part scoring up to 16 voices. A true instrument that anyone with an Apple can play.

A/D+D/A

16 channels analog to digital input. 16 channels digital to analog output. Eight bit resolution. Super-fast 8μ sec. conversion time. Monitor and output to the real world. All on one card.

EXPANSION CHASSIS

By popular demand! Eight more slots for your Apple. Attractive sturdy enclosure. Its own heavy duty power supply. Easy to use. Address cards in Expansion Chassis the same way as in your Apple. Only one additional command to specify in Apple or in Expansion Chassis. Compatible with all Apple peripherals.

MOUNTAIN HARDWARE has the most comprehensive line of Apple peripherals available. Anywere. From anybody. We know the Apple inside and out and are committed to providing the most innovative and unique products to expand and enhance its capabilities and use. After all, we were the first company to make an Apple peripheral—except Apple Computer.

The message is simple. If you have an Apple, you need to know MOUNTAIN HARDWARE.

Available at Apple Dealers worldwide.

Mountain Hardware
Leadership in Computer Peripherals
A Division of Mountain Computer, Inc.
300 Harvey West Blvd.
Santa Cruz, CA 95060 (408) 429-8600

MORE PERIPHERALS? Send me information.

NAME _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

Apple is a trademark of Apple Computer Inc.

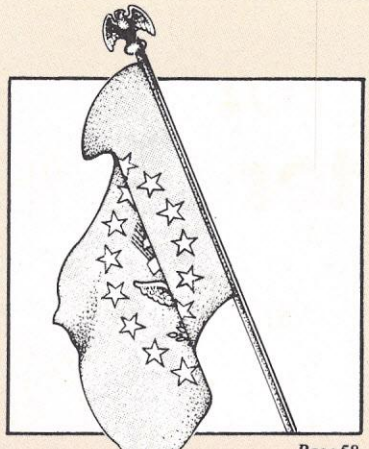
***Based on Manufacturers' published catalogs as of April 1980**

CIRCLE 3

NOVEMBER 1980 VOL. IV, NO. 11

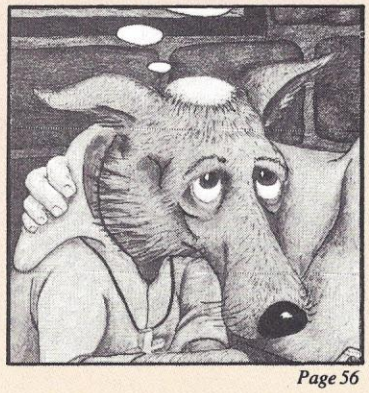
Personal Computing

For Your Home and Business



Page 58

Page 32



Page 56

Cover Illustration by Jacky Brill

DEPARTMENTS

FEEDBACK 6
RANDOM ACCESS 13
FUTURE COMPUTING ..22
COMPUTER CHESS 81
COMPUTER BRIDGE 89
PRODUCT CLOSE UP 91
WHAT'S COMING UP 95
AD INDEX 112

Computer Games will return next month.

COVER STORY

Election Night 58
Unravel the confusion surrounding election polling with this program which compiles, displays and interprets local election figures just as TV network computers do for national results. *by Nancy G. McPhee*

LAUNCHING PAD

Holiday Buyer's Guide 24
A guide to help the non-computerist select a gift for that special computerist friend. *by Ken Mazur*

Backup Deleter 38
Quickly delete your program fragments and duplicates on disks with this handy program. *by Don Wood*

Computer Cryptography for Coded Communications 66
A simple encoding and decoding process for use with your microcomputer. *by John H. Heidema*

Number Converter 74
Convert any decimal number to its equivalent in any integer number system from base 2 to base 32; or convert numbers in other systems to their decimal equivalents. *by W.B.Goldsmith, Jr.*

Novice's Notebook: Microbicide 79
Many useful hints for debugging your own programs and those taken from printed listings. *by David Lubar*

DIGGING IN

Tracking Add-On Sales 32
Follow the performance of your sales staff and individual stores with this program which provides you with a statistics report on each person and each store. *by Clint Hentz*

Translating TRS-80 Level II Basic to TI 99/4 Basic 40
A look at TI 99/4 features plus requirements and tips for making TRS-80 Level II programs run on this machine from Texas Instruments. *by Harley M. Templeton*

Free-Form Storage and Retrieval System 46
Here's a single general purpose data base program that will let you store and retrieve information in a number of different ways. *by William Lappen*

Printing Pictures from Your Pet 52
Quite simply, this program prints pictures from your Pet screen onto the Commodore 2022 tractor feed printer. *by Michael D. Gillie*

Printing and Storing Weekly Schedules 70
Toss aside that messy, unreadable appointment book; let your computer keep track of your busy schedule. *by Ivan Flores*

ON THE LIGHTER SIDE

Lamb! 56
A fable explaining one man's approach to programming. *by David Gerrold*

Publication Number ISSN 0192-5490

TBS KEEPS ON DOING IT!

The Bottom Shelf - A proud new tradition!

The fine quality software of The Bottom Shelf including its defined fields, editing features and blinking cursor off the Model I Radio Shack microcomputer has been established. Now it is time for the new. With the introduction of the Mod II, TBS went to work learning its secrets. This month will mark the introduction of major advances in business and database software for all Radio Shack computers - Mods I to III and even the little pocket computer. As this is being written, most titles are not known, but see your computer dealer or write for a catalog - all titles released are available at that time.

The one known title is *MEGAMAIL*. This is the beginning of truly usable business software for micro computers. To date programming has been hard to operate in order to work within the constraints imposed by microcomputers. Sophisticated software that is easy to use, it is a mailing system that will handle one million name lists or 8,000 name lists without skilled operators. Multi computer entry is supported. Write for details.

THE BOTTOM SHELF, INC.
P. O. BOX 49104 • ATLANTA, GA. 30359

CIRCLE 4

Personal Computing

NOVEMBER 1980 VOL. IV, NO. 11

Publisher/Editor Harold G. Buchbinder

Managing Editor Don Wood

Associate Editors Harry Shershow, Ken Mazur

Assistant Editor Marjorie Morse

Editorial Assistant Elli Holman

Contributing Editors
Thomas A. Throop, Morris Miller, George Koltanowski, William R. Parks, Evan Katz

Art Director Richard D. Sarno

Assistant Art Director Stephen C. Fischer

Art Staff
Jacky Brill, Susan N. Genser, Josh Randall, Donni Richman

Production Jon Buchbinder, Ruth Ellen Katz

Advertising Production Erik W. Slickman

Composition Manager Claudia Majetich

Composition
Florence Daniels, Robert Hildebrand, Betty Mae LaBaugh, Judy Lungelow, Maureen Riordan, Kathy Shank, Don Spencer, Harvey Sweat

Circulation Manager Dave Harriman

Circulation Elizabeth Mahon-Laidlaw

Bookstore Sales Laura J. Giuliano

General Administration
Sara Binder, Norman Jolliemore, Laurie Merrill, Malcolm Rankin, Esther Shershow

Advertising Sales. Northeast: Arlene Joseph, Benwill Publishing Corp., 2 Park Ave., New York, NY 10016 (212) 340-9780 • **California:** Michael Reynolds, 924 Westwood Blvd., Los Angeles, CA 90024; (213) 478-3017 • **Japan:** K. Yanagihara, International Business Corp., 10-10 Shinjuku 3-chome, Shinjuku-ku, Tokyo 160, Japan; (03) 350-0272; Cable Address: Yanacour Toyko.

Publisher. Published monthly by Benwill Publishing Corp.: Harold G. Buchbinder, Chief Executive Officer; George Palken, President; Domenic A. Mucchetti, Treas. Executive; Editorial and Subscription Offices: 1050 Commonwealth Ave., Boston, MA 02215. Controlled Circulation postage paid at Long Prairie, MN.

Subscription rates. U.S.: 1 year (12 issues) $14; 2 years (24 issues) $26; 3 years (36 issues) $38. Canada & Mexico: add $4/year for surface mail, $8/year for airmail. Other countries: add $8/year for surface mail, $36/year for airmail. Send subscription orders and address changes to: Circulation, Personal Computing, 1050 Commonwealth Ave., Boston, MA 02215. In Japan/Asia: K Yanagihara, International Business Corp., 10-10 Shinjuku 3-chome, Shinjuku-ku, Tokyo 160, Japan; (03) 350-0272. In Europe: Personal Computing c/o LP Enterprises, 313 Kinston Road Ilford, Essex, Eng. IG1 1PJ tel: 01 553-1001. **Back Issues.** U.S.: $3. Canada & Mexico: $4. All other countries $6.

Morgan-Grampian also publishes American City and County • Circuits Manufacturing • Contractor • Design Engineering • Digital Design • Electronics Test • Government Buyer • Industrial Distribution • Industrial Product Bulletin • Mart • Minicomputer News • Municipal Index

Audited Paid Circulation ABC

Share Your Home Programs

How do *you* use your computer at home? Family finances? Budgets? Meal planning? Entertainment? Teaching the kids? Word processing? Home security? Investment planning? Helping with your *other* hobby?

Our readers are as software-hungry as you. So why not share the home applications programs you've developed? Send us an article describing your application and the program you wrote to implement it. Be sure to include a program listing and sample run.

Remember, readers aren't familiar with your program. So explain in detail what the program does and how it does it. Include here the overall structure of your program as well as any special algorithms or routines you've used. Give suggestions for modifying or expanding the program for other applications or other situations.

All submissions should be original, typed (not all CAPS), double-spaced and neat. Include your name and address on the first page of the article and enclose a self-addressed, stamped envelope for return of material. Also, please use a fresh ribbon on your printer for program listings and sample runs.

Feel free to call us at (617) 232-5470 if you have any questions or want to discuss specific article ideas.

Mail your manuscript to:

Don Wood, Managing Editor
Personal Computing
1050 Commonwealth Ave.
Boston, MA 02215

Now NRI takes you inside the world's most popular microcomputer to train you at home as the new breed of computer specialist!

NRI teams up with Radio Shack to teach you how to use, program and service microcomputers...make you the complete technician.

It's no longer enough to be just a programmer or a technician. With microcomputers moving into the fabric of our lives (over 200,000 of the TRS-80™ alone have been sold), interdisciplinary skills are demanded. And NRI can prepare you with the first course of its kind, covering the complete world of the microcomputer.

Learn At Home in Your Spare Time

With NRI training, the programmer gains practical knowledge of hardware, enabling him to design simpler, more effective programs. And, with advanced programming skills, the technician can test and debug systems quickly and easily.

Only NRI gives you both kinds of training with the convenience of home study. No classroom pressures, no night school, no gasoline wasted. You learn at your convenience, at your own pace. Yet you're always backed by the NRI staff and your instructor, answering questions, giving you guidance, and helping you over the tough spots.

Training includes TRS-80 computer, transistorized volt-ohm meter, digital frequency counter, and the NRI Discovery Lab with hundreds of tests and experiments.

(TRS-80 is a trademark of the Radio Shack division of Tandy Corp.)

Explore the TRS-80 Inside and Out

NRI training is hands-on training, with practical experiments and demonstrations as the very foundation of your knowledge. You don't just program your computer, you introduce and correct faults ...watch how circuits interact...interface with other systems...gain a real insight into its nature.

You also build test instruments and the NRI Discovery Lab, performing over 60 separate experiments in the process. You learn how your trouble-shooting tools work, and gain greater understanding of the information they give you. Both microcomputer and equipment come as part of your training for you to use and keep.

Send for Free Catalog... No Salesman Will Call

Get all the details on this exciting course in NRI's free, 100-page catalog. It shows all equipment, lesson outlines, and facts on other electronics courses such as Complete Communications with CB, TV and Audio, Digital Electronics, and more. Send today, no salesman will ever bother you. Keep up with the latest technology as you learn on the world's most popular computer. If coupon has been used, write to NRI Schools, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI McGraw-Hill

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016
NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating including Solar Technology

Name (Please Print) Age

Street

City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council **173-110**

CIRCLE 6

All About July

Dear Sir:

After going through the July 1980 issue of your magazine I thought I just must write you a note of comments and commendation on this particular issue. I am in the business of retailing business systems and am always on the lookout for new routines for those specific applications. Now just a few comments:

Inventory Simulation. A very practical, well written program with a clear listing. Excellent samples and documentation.

Cassette Labels. Could be used by MANY, and with all of the good things about it. I have written one like this for my use too. A very usable package!

Sort. A good item to use in a large routine.

Rental Income. I loaded and modified this one. I simplified but enhanced it some. Should be very usable by many. See Figure 1 for my version.

(This was written using the new version of Microsoft Basic which permits longer variables, and in a few cases they are as long as 5 characters. However, some systems do recognize the first 2 characters of a name and this might be OK.

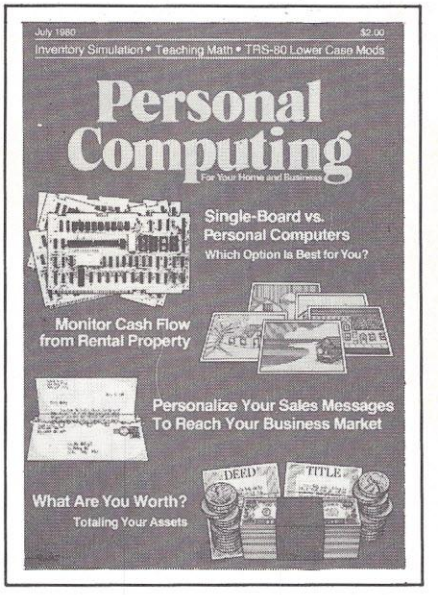
The 'LINE INPUT' is used to avoid the everpresent '?' with the regular input statement. The 'INPUT' alone would solve this.

Multi-instruction lines may be problematic for some. I am not that familiar with various systems.

The POKE 3, 1 turns off the video executed printing and the POKE 3, 3 opens the printer port as well as the video. A POKE 3, 0 would close the video and open the printer only. This is with CP/M, so other systems will again differ.

The 'GOSUB's' do eat up RAM pretty fast, but this listing requires only a little over 6K so that isn't too bad. Variables for the strings used more than once would save some RAM here.

This listing is almost 100 lines shorter than the published version, but does have several multi-instruction lines. I would be glad to correspond with anyone so deserving, if they enclose a self-addressed envelope.)



Personal Sales Messages. Very useful in many fields. The formatting of the outputs could be improved some.

Personal Assets. A very useful and simple program. The samples first cause interest otherwise possibly lost.

Fast Gomoku. Good samples. Listing is weak, but legible.

Diary. Good listing, sample and documentation. The disk handling statement could be elaborated on somewhat, or have a flow chart of the package so the individual could apply their own disk handling techniques. This has been an area costing a lot of time in using other packages on my system.

Fast Math. An interesting area of application! Should be exploited more than it is!

Ivan Denisovitch. Well . . .

Line Renumbering. A very useful feature if your operating system lacks it! Great to see this included!

Figure 1

Sample Run

```
--**  S T A T E M E N T   O F   R E N T A L   P R O P E R T Y  **--
================================================================
                    (Less depreciation)

                           FOR
                                          S.S. NO. :

Description of Property :
Address  of  Property :  ,

              --**  I N C O M E  **--
              ---------------------------

RENTS  RECEIVED                          $   200.00
LAYINGTILE                               $    40.00
                                         ------------
        Gross Income                                    $   240.00

          --**  E X P E N S E S  **--
          ------------------------------

     Advertising                         $     1.00
     Cleaning                            $     2.00
     Gardening                           $     3.00
     Insurance                           $     4.00
     Interest                            $     5.00
     Janitor & Heat                      $     6.00
     Legal & Accounting                  $     7.00
     Management                          $     8.00
     Office Supplies                     $     9.00
     Salaries                            $    10.00
     Supplies                            $    11.00
     Taxes & Licenses                    $    12.00
     Telephone                           $    13.00
     Utilities                           $    14.00
     Automobile :
      15 miles at  12  cents / mile.     $     1.80
     Other Travel                        $    16.00
     Repairs :
          LANU                           $    20.00
     Other Expenses :
          ROOF                           $    21.00
                                         ------------
       Gross Expenses                                $   163.80

       Cash Flow from Property                       $    76.20
                                                     ==========
```

Sample Run continued

```
  --**  S T A T E M E N T  O F  R E N T A L  P R O P E R T Y  **--
=====================================================================
                    (Less depreciation)

                  FOR June 1 to July 1, 1980
   Robert T. Wilson                          S.S. NO. :  123-45-6789

   Description of Property : Multiplex # 1
   Address   of   Property : 555 NE Sycamore Dr.  Sand City, OR 98765

                    --**  I N C O M E  **--
                   ----------------------------

      RENTS  RECEIVED                          $ 6,558.85
      Plot Plan Layout                         $   124.50
      Construction Plans                       $    84.10
                                               ----------
            Gross Income                                    $ 6,767.45

              --**  E X P E N S E S  **--
             ---------------------------------

            Advertising                        $    51.20
            Cleaning                           $   253.15
            Gardening                          $    27.50
            Insurance                          $   635.45
            Interest                           $   545.65
            Janitor & Heat                     $    51.50
            Legal & Accounting                 $    25.50
            Management                         $ 1,225.75
            Office Supplies                    $   111.25
            Salaries                           $    51.85
            Supplies                           $    54.65
            Taxes & Licenses                   $   321.85
            Telephone                          $    15.88
            Utilities                          $    35.85
            Automobile :
             145 miles at  12  cents / mile.   $    17.40
            Other Travel                       $    15.63
            Repairs :
                 Roof                          $    25.45
                 Gutters                       $    15.85
                 Yardlights                    $    23.65
                 Windows                       $    51.45
                 Painting                      $   115.25
            Other Expenses :
                 Lawn Planting                 $    25.15
                                               ----------
             Gross Expenses                                 $ 3,696.86

             Cash Flow from Property                        $ 3,070.59
                                                            ==========
```

Program Listing

```
10      'RENTALS'   INCOME AND EXPENSES OF RENTAL PROPERTY  -------
20      'Requires a little over 6K of RAM ----------
30      'Revised from 'Personal Computing'  July 1980, page 40 - 42 -----
40      'Written by W.B. Goldsmith, Jr. -------------
50 DIM R$(20),E(35),RR(10),IA(10),EX(18)
60 LI =0:D=15:R=45:CL$=CHR$(26):PRINT CL$       'CL$ clears the screen ---
70 DIGITS = 2   'Not required by some systems ------
80 V$= "$ ####,.##"

90      '-------------------------------------------------<>
100     '-----  P R O P E R T Y   D A T A  E N T R I E S  --------<>
110     '-------------------------------------------------<>

120 H$="Statement of Rental Income and Expenses":GOSUB 1370:GOSUB 1390:PRINT
130 H$="This Program will collect data and print a summary for one property"
    :GOSUB 1370:PRINT
140 H$="Use Multiple runs for other properties.":GOSUB 1370:GOSUB 1380
150 H$="Enter Owners Name":GOSUB 1400:LINE INPUT A$(1)
160 H$="Enter Owners ID/SS Number":GOSUB 1400:LINE INPUT A$(2)
170 H$="Enter Kind of Property":GOSUB 1400:LINE INPUT A$(3)
180 H$="Enter Street Address":GOSUB 1400:LINE INPUT A$(4)
190 H$="Enter City":GOSUB 1400:LINE INPUT A$(5)
200 H$="Enter State":GOSUB 1400:LINE INPUT A$(6)
210 H$="Enter ZIP code":GOSUB 1400:LINE INPUT A$(7)
220 H$="Enter Period of time covered":GOSUB 1400:LINE INPUT A$(8):PRINT

230     '-----------------------------------------<>
240     '-------  I N C O M E  E N T R I E S  ----------<>
250     '-------------------------------------------------------<>

260 H$="---***  E N T R I E S  O F  R E N T S  ***---":GOSUB 1370:GOSUB
    1390:PRINT
270 H$="Up to 10 rent entries - enter 'Return' to break out ...":GOSUB 1370
    :GOSUB 1390:PRINT
280 FOR A = 1 TO 10
290 H$="Enter Rents Received":GOSUB 1410:LINE INPUT Z$:RR(A)=VAL(Z$)
300 IF LEN(Z$)=0 THEN 320
310 RT=RT+RR(A):PRINT:NEXT
320 PRINT:GOSUB 1390:PRINT
330 H$=" Up to 10 income sources - Enter 'Return' to break out ...":GOSUB 1370
    :GOSUB 1390:PRINT
340 FOR B= 1 TO 10
350 H$="Enter other income source":GOSUB 1400:LINE INPUT IC$(B)
360 IF LEN(IC$(B))=0 THEN B=B-1:PRINT:GOTO 410
370 IF ASC(IC$(B))<65 THEN 380 ELSE 390
380 H$="Please enter proper income source ..."
    :GOSUB 1370:GOTO 350
390 H$="Enter income amount":GOSUB 1410:LINE INPUT IA$:IA(B)=VAL(IA$)
400 IT=IT+IA(B):NEXT
410 GI=IT+RT            'Totals rent and Income into Gross Income ----

420     '-----------------------------------------<>
430     '-----  E X P E N S E S  --------------------<>
440     '-------------------------------------------------------<>

450 H$="---***  E X P E N S E S  ***---":GOSUB 1370:GOSUB 1390:PRINT
460 H$="Enter expenses as prompted . . .":GOSUB 1370:PRINT
470 H$="Advertising":GOSUB 1410:LINE INPUT ADV$:EX(1)=VAL(ADV$):PRINT
480 H$="Cleaning":GOSUB 1410:LINE INPUT CLE$:EX(2)=VAL(CLE$):PRINT
490 H$="Gardening":GOSUB 1410:LINE INPUT GAR$:EX(3)=VAL(GAR$):PRINT
500 H$="Insurance":GOSUB 1410:LINE INPUT INS$:EX(4)=VAL(INS$):PRINT
510 H$="Interest":GOSUB 1410:LINE INPUT INR$:EX(5)=VAL(INR$):PRINT
520 H$="Janitor & Heat":GOSUB 1410:LINE INPUT JH$:EX(6)=VAL(JH$):PRINT
530 H$="Legal & Accounting":GOSUB 1410:LINE INPUT LA$:EX(7)=VAL(LA$):PRINT
540 H$="Management":GOSUB 1410:LINE INPUT MAN$:EX(8)=VAL(MAN$):PRINT
550 H$="Office Supplies":GOSUB 1410:LINE INPUT OS$:EX(9)=VAL(OS$):PRINT
560 H$="Salaries":GOSUB 1410:LINE INPUT SAL$:EX(10)=VAL(SAL$):PRINT
570 H$="Supplies":GOSUB 1410:LINE INPUT SUP$:EX(11)=VAL(SUP$):PRINT
580 H$="Taxes & Lisenses":GOSUB 1410:LINE INPUT TL$:EX(12)=VAL(TL$):PRINT
590 H$="Telephone":GOSUB 1410:LINE INPUT TEL$:EX(13)=VAL(TEL$):PRINT
600 H$="Utilities":GOSUB 1410:LINE INPUT UTI$:EX(14)=VAL(UTI$):PRINT
610 H$="Auto Mileage":GOSUB 1400:LINE INPUT CAR$:EX(15)=VAL(CAR$):PRINT
620 CA=12:CT=CA*EX(15)/100
630 H$="Other Travel":GOSUB 1410:LINE INPUT OT$:EX(16)=VAL(OT$):PRINT
640 FOR C = 1 TO 14:ET=ET+EX(C):NEXT:PRINT TAB(Q)"Total of Expenses  $ ";ET
    ;PRINT:GOSUB 1420:PRINT
650 H$="Enter Repairs - up to 10 - 'Return' to break out ...":GOSUB 1370
    :GOSUB 1390:PRINT
660 FOR D = 1 TO 10
670 H$="Enter type of repair":GOSUB 1400:LINE INPUT RE$(D)
680 IF LEN(RE$(D))=0 THEN D=D-1:GOTO 710
690 H$="Enter cost of repair":GOSUB 1410:LINE INPUT RC$(D)
700 PT=PT+VAL(RC$(D)) : NEXT
710 PRINT:PRINT TAB(20) STRING$(40,45):PRINT
720 H$="Enter other expenses - up to 10 - 'Return' to break out ...":GOSUB 1370
    :GOSUB 1390:PRINT
730 FOR E = 1 TO 10
740 H$="Enter Other Expense":GOSUB 1400:LINE INPUT OE$(E):PRINT
750 IF LEN(OE$(E)) = 0 THEN E=E-1:GOTO 780
760 H$="Enter amount of Expense":GOSUB 1410:LINE INPUT EX$(E)
770 EX=EX+VAL(EX$(E)):NEXT
780 GE=ET+PT+EX+EX(16)+CT               'DETERMINES GROSS EXPENSES ------
790     '-------------------------------------------<>
800     '-----  R E P O R T  P R I N T  O U T  -----------<>
810     '-------------------------------------------------------------<>

820 Q=20:R=55
830 H$="Hard copy or Screen Report ? (H/Cr=S) ":GOSUB 1400:LINE INPUT TT$
840 IF TT$<>"H" THEN 860
850 H$="Printer ready ? (Return) ":GOSUB 1400:LINE INPUT EE$:PRINT CL$:POKE 3,3
    : FOR Q1 = 1 TO 4:PRINT:NEXT
860 H$="--**  S T A T E M E N T  O F  R E N T A L  P R O P E R T Y  **--"
    :GOSUB 1370:GOSUB 1390:R=55
870 H$="(Less depreciation)":GOSUB 1370:PRINT
880 H$="FOR "+A$(8):GOSUB 1370
890 PRINT TAB(8) A$(1) TAB(52)"S.S. NO. : " TAB(64) A$(2):PRINT
900 PRINT TAB(8)"Description of Property : " A$(3)
910 PRINT TAB(8)"Address   of   Property : "A$(4)"  "A$(5)", "A$(6)" "A$(7)
    :PRINT
920 H$="--**  I N C O M E  **--":GOSUB 1370:GOSUB 1380:PRINT
930 PRINT TAB(12)"RENTS  RECEIVED";:PRINT TAB(55)USING V$;RT
940 FOR F = 1 TO B
950 PRINT TAB(12) IC$(F);:PRINT TAB(R) USING V$; IA(F)
960 NEXT
970 PRINT TAB(R) STRING$(12,45)
980 PRINT TAB(22)"Gross Income";:PRINT TAB(65)USING V$;GI
990 PRINT:H$="--**  E X P E N S E S  **--":GOSUB 1370:GOSUB 1380:PRINT
1000 T=20:U=55
1010 PRINT TAB(T)"Advertising";:PRINT TAB(U)USING V$;EX(1)
1020 PRINT TAB(T)"Cleaning";:PRINT TAB(U)USING V$;(EX(2))
1030 PRINT TAB(T)"Gardening";:PRINT TAB(U) USING V$;EX(3)
1040 PRINT TAB(T)"Insurance";:PRINT TAB(U)USING V$;EX(4)
1050 PRINT TAB(T)"Interest";:PRINT TAB(U)USING V$;EX(5)
1060 PRINT TAB(T)"Janitor & Heat";:PRINT TAB(U)USING V$;EX(6)
1070 PRINT TAB(T)"Legal & Accounting";:PRINT TAB(U)USING V$;EX(7)
1080 PRINT TAB(T)"Management";:PRINT TAB(U)USING V$;EX(8)
1090 PRINT TAB(T)"Office Supplies";:PRINT TAB(U)USING V$;EX(9)
1100 PRINT TAB(T)"Salaries";:PRINT TAB(U)USING V$;EX(10)
1110 PRINT TAB(T)"Supplies";:PRINT TAB(U)USING V$;EX(11)
1120 PRINT TAB(T)"Taxes & Licenses";:PRINT TAB(U)USING V$;EX(12)
1130 PRINT TAB(T)"Telephone";:PRINT TAB(U)USING V$;EX(13)
1140 PRINT TAB(T)"Utilities";:PRINT TAB(U)USING V$;EX(14)
1150 PRINT TAB(T)"Automobile :"
1160 PRINT TAB(T)EX(15)"miles at ";CA;" cents / mile.";:PRINT TAB(U)USING V$;CT
1170 PRINT TAB(T) "Other Travel";:PRINT TAB(U)USING V$;EX(16)
1180 PRINT TAB(T)"Repairs :"
1190 FOR P = 1 TO D
1200 PRINT TAB(T+5);RE$(P);:PRINT TAB(U)USING V$;VAL(RC$(P)):NEXT
1210 IF E=0 GOTO 1250
1220 PRINT TAB(T) "Other Expenses :"
1230 FOR S= 1 TO E
1240 PRINT TAB(T+5);OE$(S);:PRINT TAB(U)USING V$;VAL(EX$(S)):NEXT
1250 PRINT TAB(R) STRING$(11,45)
1260 PRINT TAB(22)"Gross Expenses";:PRINT TAB(65)USING V$;GE
1270 CFLOW=GI-GE:PRINT  'Cash flow calculation -----
1280 PRINT TAB(22)"Cash Flow from Property ";:PRINT TAB(65)USING V$;CFLOW
1290 PRINT TAB(63) STRING$(14,61)
1300 POKE 3,1:R=65:ET=0
1310 H$="Another copy of the report ? ('Return'/N)":GOSUB 1400:LINE
     INPUT RR$
1320 IF RR$="N" THEN R=45: GOTO 120
1330 GOTO 830
1340    '-----------------------------------------<>
1350    '-------  AUTOCENTER SUBS  -------------------------<>
1360    '-------------------------------------------------------------<>

1370 PRINT TAB((78-LEN(H$))/2);H$:RETURN  'Autocenter of H$
1380 PRINT TAB(((78-LEN(H$))/2)-2);STRING$(LEN(H$)+4,45):RETURN 'UNDRLINE W/----
1390 PRINT TAB(((78-LEN(H$))/2)-2);STRING$(LEN(H$)+4,61):RETURN 'UNDRLINE W/====
1400 PRINT TAB(Q);H$;TAB(R);": ";:RETURN  'AUTO TAB FOR INPUT W/ ": "
1410 PRINT TAB(Q);H$;TAB(R)"$ ";:RETURN  'AUTO TAB FOR INPUT W/ "$ "
1420 PRINT:PRINT TAB(20) STRING$(40,45):PRINT:RETURN
```

Amort Tables. A very popular item with investment brokers! This version is well presented too.

Video Tape Selector. Could be used for a floppy directory too, possibly. Very useful!

Save Bytes. Excellent asset to programmers. Two versions of the listing makes it better!

Data Management Made Easy. Sounds like a great article. I have not completely waded though this one either, but could be a lot of help in this area.

So in this issue there are over a dozen very useful and practical well written articles with listings. I think this is fantastic. I have seen in another recent magazine a lot of games and similar useless packages that lose their effectiveness shortly after you get them loaded and debugged! But these can be in use for a long time!

Thanks a lot for such a good publication, and keep it going so all of those keypunchers out there have something to use in their CPU's.

Lloyd F. Bazant
4714 S.W. Willetta St.
Albany, OR 97321

Software Needed for Sord Mark II

Dear Editors:

We've just acquired a Sord Mark II Model 223 microcomputer for use in our subscription fulfillment operation and are looking for somebody who can supply us software covering general ledger, accounts payable and purchasing on an off-the-shelf basis.

The Sord uses Extended Basic and is manufactured by Sord Computer Systems, Inc., Isoma No. 2 Bldg., 42-12 Nishi-Shikoiwa 4-chome, Katsushika-ku, Tokyo, Japan 124.

Also (for fun) I'd like to find a program to play checkers, backgammon and cribbage with my Sord (I'll probably spend evenings at the office on that basis).

The Sord is relatively new to this country, but is a powerful unit with a 64K memory and expansion capabilities. Perhaps some of the other Sord purchasers have developed or adapted other software to fit it.

Lauren R. Januz
President
Januz Marketing Communications Inc.
Evergreen 41 Executive Center
49 Sherwood Terrace
Lake Bluff, IL 60044

Computer Carousel Projector

Dear Editor,

In reference to the article "Getting Your Act Together" by Cecil Smith, which appeared in the June 1980 issue, we would like to point out that for the past year and a half, Computer Action has been using a more automated system to accomplish a similar task.

Computer Action has developed an interface between the Kodak carousel projector and the Radio Shack TRS-80 microcomputer, for use with our educational software, and can easily be adapted to many other micros.

This interface enables the computer to automatically select and position the projectors' carousel, thus displaying the proper slide. A small software routine is used to calculate the correct direction for quick access to a slide.

We feel that this type of an interface would be an asset to anyone considering the implementation of Mr. Smith's program. Computer Action will be happy to supply, at no charge, the schematics and instructions for the construction and operation of the interface to any interested reader sending us a legal size stamped self-addressed envelope.

Lance Sprung
Computer Action
45 Paerdegat 2nd St.
Brooklyn, NY 11236

Basic Typist

Dear Editors:

I realize that "Basic Typist" (April *PC*) is meant to be used with "Basic Typesetter" (May *PC*) but it can be used alone more easily if the following facility is added to it:

```
1002 FOR X = 1 TO L:PRINT
   X;:;PRINTA$(X):NEXT
```

The above line provides the numbered lines every time you return to adding text from some other function. Another aid would be the addition of line numbers in the margin of the video text. This can be accomplished by simply initiating line 1005 with:

```
1005 PRINTL+1;:
```

I am a newcomer to this fascinating hobby. It is like having a continent to explore in your own living room.

Eli Passin
New York, NY

Improving A Handy Sort

Dear Sir:

I found Dwight Wheeler's "A Handy Sort" (July 1980) a reasonably good exposition of the bubble sort. There is one area, however, where it may be improved.

As one proceeds through the items to be sorted, it is very easy and useful to use a flag to indicate whether or not any items have been "swapped." If items are *not* in order at least one swap will occur; if they are in order no swaps will be needed. To implement this, consider the following steps:

1. 172 REM — SET SWAP FLAG
   174 F=0
   176 REM — 0=OFF/1=ON
2. At line 190 a check is made to see if the items are in the desired order. If they are, control is transferred to line 260, otherwise a swap is performed. Therefore:
   195 F=1
3. We need to know, after passing through that subset of the items which needed sorting, if we have swapped any items. If we have, we must make at least one more pass. If we have not, we are through. Therefore:
   265 IF F=0 THEN 272

What difference does all this make? Well, in Mr. Wheeler's example he uses 20 data items. He must, therefore,

Suddenly, RCA makes talking to your computer a lot cheaper.

New interactive data terminal with color graphics–only $369.*

RCA's new VP-3301 is a professional quality, ASCII encoded, interactive data terminal, suitable for a wide variety of industrial, educational, business and individual applications requiring interactive communication between computer and user. Connects directly to your computer or to a standard modem for over the phone access to time sharing networks and data bases. And it's compatible with networks such as those provided by CompuServe Information Services and Source Telecomputing Corp. Microprocessor intelligence and LSI video control integrated circuits bring performance, features and flexibility at a low price. Operates from 5 volt power supply (included).

Unique color locking circuitry creates sharp, jitter free, true color graphics and rainbow free characters.

You can display the entire field of characters in any of 8 colors against any of 8 background colors (8 gray scales with monochrome monitors). Or to add special emphasis, you can display individual letters, words or lines in different colors or in reverse video.

The VP-3301 offers you a choice of two software-selectable display formats: Either 40 characters by 24 lines. Or 20 characters by 12 lines.

The terminal's resident character set consists of 52 upper and lower case alphabetics, 10 numerals, 32 punctuation/math symbols and 31 control characters.

You can also define a total of 125 of your own characters. Including: Greek letters and other foreign alphabets, graphic symbols, large graphics building blocks, playing card suits, unique character fonts and "little green men."

The keyboard section features flexible-membrane key switches with contact life rated at greater than five million operations. A finger positioning overlay and positive keypress action give good operator "feel".

An on-board sound generator and speaker provides aural feedback for key presses and may also be activated with escape sequences to provide an audio output.

The sealed keyboard surface is spill proof and dust proof. This, combined with high noise immunity CMOS circuitry, makes the VP-3301 ideal for hostile environments.

Output is industry standard asynchronous RS232C or 20 mA current loop with 6 switch-selectable baud rates and 8 selectable data formats.

You can connect the terminal directly to a 525 line color or monochrome monitor. Or to a standard TV set using your RF modulator.

For more information, contact RCA MicroComputer Marketing, New Holland Avenue, Lancaster, PA.

Or call our toll-free number: 800-233-0094.

*Suggested user price.

RCA

CIRCLE 7

execute lines 180-260 no fewer than 189 times no matter what order the original items are in. His algorithm would do so even if they were already in perfect order. Mine would make only one pass if the items were in perfect order.

Michael C. Harris
Baltimore, MD

Handshaking Resumes

Dear Editor:

In the June issue of *Personal Computing*, you published my letter concerning the non-handshaking situation between my Heath H14 printer and my NorthStar Horizon serial interface. Thanks to your audience, I was deluged with letters as well as phone calls.

Software solutions outnumbered hardware solutions about two-to-one, and the simplest hardware solution was provided by Al Weiner of Waltham, MA. Without rewiring the Horizon header at 4D, all I had to do was change the Heath end of the cable to move DB25 connector pin 20 to Heath connector pin 4 to tap the NOT-RTS signal provided in the Heathkit modification on DB25 connector pin 15. Al even provided a three-line Basic program to prove that the handshaking was working at 4800 baud!

At first, the solution didn't work. I finally hooked up my TVOM and discovered I didn't have the NOT-RTS signal at the printer (although I had followed the modification instructions during assembly). I had never opened the printer since its first visit to Heathkit Electronics Center in Seattle and assumed that they had checked it out thoroughly.

They certainly did! Even though I advised them of the modification number, they ripped out the additional wiring underneath the PC board! I've rewired the mod, and my H14 is purring along at 4800 Baud with full handshaking under CP/M as well as N*DOS. The fix would have been implemented two weeks ago had HEC told me of their action months ago.

Thanks to everybody who replied to my previous letter to the editor — it's nice to know I'm not alone in the wilderness without help! And thank you, *Personal Computing*, for publishing my complaint which triggered all the response.

John R. Dye
Lacey, WA

Saving Bytes

Dear Sir:

I had just finished writing the following routine for saving bytes with dates, when I received your July issue. My routine is a little different than Mr. Tzinberg's ("Save those Bytes") and though it takes four bytes to save, a few less bytes are required in the program. Perhaps others may be interested in this version as well.

I did try multiplying the variable A by 10000 to get an integer, but for dates after March 26, an overflow error will result as the integer value of 32767 is exceeded. However, the single precision cost of four bytes is still a big help in fielding some random files.

```
100 INPUT "ENTER THE DATE (MM/
    DD/YY) ";DT$:IF LEN (DT$)<>8
    THEN PRINT"REDO": GOTO 100
110 D1$=".":D2$=" ":MID$
    (DT$,3)=D1$:MID$(DT$,6)=
    D2$:A=VAL(DT$)
5000 DT$=STR$(A):B=INSTR(DT$,"."):
     IF LEN(DT$)<8 THEN DT$=DT$+
     "00"
5010  MO$=STR$(INT(A)):DA$=MID$
      (DT$,B+1,2):YR$=MID$(DT$,B
      +3,2):DT$=MO$+"/"
      +DA$+"/"+YR$
```

Line 100 asks for the data and insures that it is in the proper format. In line 110, the first / (slash) is replaced with a decimal point (D1$); the second / (slash) is replaced with a space (D2$). The decimal point is necessary to avoid confusion that may arise with dates like 01/02/80. The resultant number without a decimal would be 1028; either a 10/2/80 or a 1/02. A 1.028 number makes it certain. The variable A can now be saved.

In lines 5000 and 5010, the date string is reconstructed. The decimal point is used to determine / (slash) placement.

Paul Reinertson
Bangkok, Thailand

Keeping Disks Clean

Dear Editor:

Regarding your recent (September 1980) comments in Random Access concerning potential volcanic ash damage to floppy disks, I'd like to suggest a protective means which has worked well in my own system.

Living in the city and having a cat for a second pet (my Apple being first pet, of course) means lots of dirt and dust. To prevent infiltration of my diskettes, I keep each one, in its standard envelope, in a carefully sealed Dow Ziploc brand sandwich bag. This bag is characterized by an air-tight seal that can be opened and closed manually many times. And the bags are very inexpensive.

The one quart 6-5/8″ × 5-7/8″ size is perfect for my 5-1/4″ diskettes. One possible drawback is the inability to fit a diskette so protected into the clear plastic 3-hole binder files which are so popular. Instead, I use the larger 8″ floppy disk pages. I find that several diskettes, each individually packaged in its own Ziploc brand bag, will fit into a large pocket. For example, I keep three or four related files together, like my Applewriter floppies.

I've been using this technique for some time, without problems of any kind. My diskettes load and run fine, each time. Of course, I also perform other preventative maintenance, like cleaning the disk drive heads with the new 3M kit and keeping the speed right at 300 rpm with Disco-Tech's disk drive timing program.

But the bottom line has been reliable system operation in a non-optimum working environment.

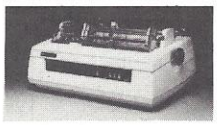
Keep up your good work with *Personal Computing*!

Harry E. Brawley, Jr.
Cambridge, MA

WE WILL NOT BE UNDERSOLD

16K MEMORY UPGRADE KITS 2 for $85 **$45**
for TRS-80*, Apple II, (specify): **Jumpers** **$2.50**

PRINTERS



NEC Spinwriter
Letter Quality High Speed Printer
Includes TRS-80* interface software, quick change print fonts, 55 cps, bidirectional, high resolution plotting, graphing, proportional spacing: R.O. **$2550**
R.O. with Tractor Feed **$2650** KSR with Tractor Feed **$2950**

779 CENTRONICS TRACTOR FEED PRINTER **$969**
Same as Radio Shack line printer I

737 CENTRONICS FRICTION & PIN FEED PRINTER **$799**
9 x 7 matrix Same as Radio Shack line printer IV

730 CENTRONICS FRICTION & PIN FEED PRINTER **$629**
7 x 7 matrix Same as Radio Shack line printer II

P1 CENTRONICS PRINTER Same as Radio Shack quick printer **$269**

PAPER TIGER (IP440) Includes 2K buffer and graphics option **$879**
(IP460) Bidirectional, 160 cps, graphics and 2K buffer **$1075**

TI-810 Faster than Radio Shack line printer III. Parallel and serial w/TRS-80* interface software w/u + I case & paper tray **$1589**
Compressed print, vertical form control **$1865**

OKIDATA Microline 80 Friction and pin feed **$549**
Tractor Feed, friction, and pin feed **$649**
Microline 83 Bidirectional, 120 cps, uses up to 15" paper **$1050**

EATON LRC 7000 + 64 columns, plain paper **$299**

ANADEX **DP-9500** **$1359** **DP-8000** **$825**

CAT MODEM Works same as Radio Shack Telephone Interface II **$148**
LEEDEX MONITOR Video 100 **$119**
ZENITH Color Monitor **$379**
SANYO Model VM 4509 9" Monitor **$155**

DISK OPERATING SYSTEMS

PATCHPAK #4 by Percom Data **$ 8.95**
CP/M® for Model I, Zenith **$145** • for Model II, Altos **$169.00**
NEWDOS Plus **40track** **$ 99.00**
NEWDOS 80 **$135.00**

ACCESSORIES

HEAD CLEANING DISKETTE: Cleans drive Read/Write head in 30 seconds. Diskettes absorb loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5¼" or 8". **$20 ea/$45 for 3**

FLOPPY SAVER: Protection for center holes of 5¼" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool, tough 7-mil mylar reinforcing rings. Installation tools and rings for 25 diskettes. **$ 11.95** **Re-orders of rings only** **$ 7.95**

EXTERNAL DATA SEPARATOR: Eliminates data separation problems (crc). Improves reliability. This plug in unit comes fully assembled and tested. **$ 29.95**

RS232 **$ 84.00**

TRS232: Teletype current loop output from cassette port **$ 49.00**

DISK-DRIVE EXTENDER CABLES: Fits all mini-disk drives. **$ 16.95**

SIX (6) PRONG ISOLATOR: ISO-2 **$ 54.00**

AC FILTER/6 PRONG POWER STRIP **$ 39.00**

DISK DRIVE CABLES: 2 drive **$29.00** 4 drive **$ 35.00**

DUST COVERS: TRS-80/Apple **$ 7.95**

PLASTIC DISKETTE HOLDER: For ring binder, holds 20 **$ 8.00**

RF MODULATOR: Adapts video to TV **$ 35.00**

TRS-80 & OTHER MYSTERIES **$ 18.95**

NEC SPINWRITER THIMBLE **$19.95** **RIBBON** **$ 6.95**

CCS CARDS: Parallel or serial **$115.00**

DISK DRIVES **$314**

40 track, 102K Bytes. Fully assembled and tested. Ready to plug-in and run the moment you receive it. Can be intermixed with each other and Radio Shack drive on same cable. TRS-80* compatible silver enclosure. 90 day warranty. One year on power supply. **External card edge included.**

FOR TRS-80*		
CCI-100	5¼", 40 Track (102K Bytes) for Model I	**$314**
CCI-280	5¼", 80 Track (204K Bytes) for Model I	**$549**
CCI-800	8" Drive for Model II (½ Meg Bytes)	**$795**
For Zenith Z89		
CCI-189	5¼", 40 Track (102K Bytes) add-on drive	**$394**
Z-87	Dual 5¼" add-on drive system	**$995**

DISKETTES — Box of 10 (5¼") - with plastic library case **$24**
8" double density for Model II (box of 10) **$36**

COMPLETE SYSTEMS

ALTOS 64K, DD, SS, 2-Drive, 1MB **$3995**
APPLE 16K **$989**
TRS-80* Model II-64K **$3499**
TRS-80* LEVEL II-16K with keypad **$689**
TRS-80* Expansion Interface **$249**
HEWLETT PACKARD HP-85 **$2950**
ZENITH Z89, 48K all-in-one computer **$2500**
ZENITH Z19 **$735**
TELEVIDEO **912B $745** **912C $755** **920B $769** **920C $779**
ATARI 400 $489 **ATARI 800 $749**
APF **Game Only $99** **Complete System $499**
MATTEL INTELLIVISION **$229**

Software available for all Complete Systems

SOFTWARE FOR THE TRS-80* Software w/Manual

INTELLIGENT TERMINAL SYSTEM ST-80-III BY LANCE MIKLUS: Enables a TRS-80* to act as a dial-up terminal on any standard time sharing network. Provides a TRS-80* with control key, ESC Key, Repeat Key, Rub Out Key, Break Key, full upper and lower case support, selectable printer output and program selectable transmission rates **$139**

CCA-DATA MANAGEMENT SYSTEM: Automate your information processing tasks. You can create a file of customer information; quickly and easily add, delete or update records; search a file; keep a file in order of the value in any field; and print records and labels in any desired sequence or from just a part of a file. Requires 32K TRS-80 and one drive. **$72.00**

S & M SYSTEMS
INSEQ-80™: Indexed Sequential Access Method (ISAM) for the TRS-80 Model I. A must for anyone writing business programs. Eliminate wasted disk space from direct record processing. Split second access to any record. Access data records instantly via alpha/numeric "key" eg. Part NR, zip code or sequentially in ascending key sequence. Add/modify records in any order. Access up to three files per program — Files may be spread over multiple disks. Machine language processing from your basic program. Utility program to convert direct files to INSEQ-80 format. **$49.00**

FULLY INTERACTIVE ACCOUNTING PACKAGE: ISAM (INSEQ-80) based. Includes General Ledger, Accounts Payable, Accounts Receivable and Payroll. System runs "stand alone" or "co-ordinated G/L" at users option. Based on Osborne accounting method. Requires 32K, TRS-80, 2 or 3 drives. N/A CA.

General Ledger	**$99**
Accounts Receivable	**$99**
Accounts Payable	**$99**
Payroll	**$99**

Osborne books: Req'd as additional documentation $20 ea

INVENTORY Requires 32K, TRS-80, 1 drive **$125**
INSORT-80: Callable form BASIC via USR. Sorts "Random" Disk Files. "Disk" to "Disk" sort times — 350 records in 35 secs, 1000 records in 6 minutes, 3500 records in 12 minutes. Machine language processing. Up to 35 sort keys ascending/descending. Utility to build BASIC program. Runs under NEWDOS. **$49.95**

CP/M® BASED SOFTWARE for Zenith, Altos, Radio Shack, Apple Software w/Manual

Z-80 SOFTCARD FOR APPLE: Your key to future software expansion. Get the best of both worlds, Apple's 6502 and CP/M Z-80. Plug in the card and get a Z80. Supports Apple language card and all Apple peripherals. Comes with set of three manuals. **$339**

CCI-TELNET VERSION 5: A communication Package which enables microcomputer users to communicate both with Large Mainframes and other microcomputers. Extensive commands make it useful in many applications where communication between computers is necessary. Powerful terminal mode enabling user to save all data from a session on disk. Completely CP/M compatible. Multiple communication protocols supported. Able to transfer files in both directions without protocol where the other machine does not support any protocol. Extensive ON-SCREEN help. Source code provided. **$149**

MICROPRO-WORD-STAR: Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace. Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning. **$399**

DEALER (NATIONAL/INTERNATIONAL) INQUIRIES INVITED **Send for FREE Catalogue**

The CPU SHOP

TO ORDER CALL TOLL FREE 1-800-343-6522
Massachusetts residents call **(617) 242-3361**
For detailed technical information, call 617/242-3361
Hours: 10AM-6PM (EST) M-F (Sat. till 5)

5 Dexter Row, Dept. PC11
Charlestown, Massachusetts 02129

Massachusetts residents add 5% sales tax

Quantities on some items are limited

VISA®

master charge

*TRS-80 is a Tandy Corporation Trademark ® Digital Research

CIRCLE 9

RANDOM ACCESS

An Alternative to Commuting

Telecommuting. The word may not be part of most people's vocabularies today, but it probably will be within a few years.

In fact, by 1990, people won't just be talking about telecommuting, says Jacks Nilles, the University of Southern California researcher who coined the word. Millions of people will be doing it, five days a week or more. And those who aren't might be sporting bumper stickers proclaiming "I'd rather be telecommuting."

"Explained most simply, telecommuting is the substitution of telecommunications and computers for the commute to work," says Nilles, whose research on alternative work patterns is described in Alvin Toffler's bestselling book, "The Third Wave."

Two types of telecommuters will evolve, predicts Nilles, who directs the interdisciplinary programs at USC. Some will eliminate the workday commute entirely, turning portions of their homes into offices. Others, who need face-to-face contact to do their jobs, will shorten their commutes by traveling to nearby regional offices that are tied by telecommunication systems to central headquarters many miles away.

Converting a room at home into a telecommunications station will not be particularly expensive, Nilles says, since it will require only the addition of a personal computer terminal and a modem.

Telecommuters most likely to stay home, he says, will be specialists who don't usually need face-to-face contact to do their work — researchers, and lawyers not often involved in court proceedings.

Some workers may combine the two types of telecommuting, going to a regional office for everyday work and staying home occasionally to complete reports and other "solo" work.

Some families may telecommute together, with one parent traveling to a regional office, the other working at home on a computer connected to the office, and the children in charge of proofreading and data storage tasks.

Telecommuting will not become the norm, Nilles says, but it will become a viable work pattern, based on the premise that it's easier and cheaper to move information than to move people. Telecommuting will become feasible for about half the labor force — the workers whose jobs deal mainly with information.

"By 1990, an estimated 47 million people will have information jobs in the United States," says Nilles. "Maybe 2 to 5 million of those people will be telecommuting to some degree."

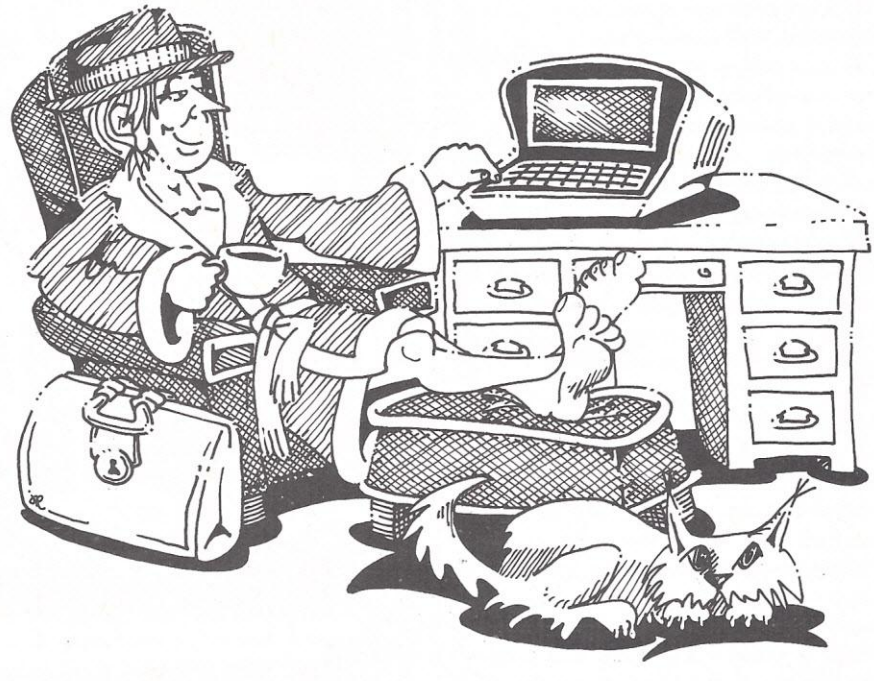
Besides having more leisure time, gained by eliminating or reducing the daily commute, telecommuters will enjoy a substantial savings in fuel costs.

In a two-year study funded by the National Science Foundation, Nilles and his research team found that the 2,000 employees of a large Los Angeles-based insurance company averaged 21.4 miles per day commuting. Nilles figures that, for every seven employees who telecommute, the annual energy saving would equal the electrical energy required to run an average American home for a year.

Freeway-filled Los Angeles is not the only place where telecommuting will work. "We did a survey of major cities to find out how far people commuted to work," Nilles says. "The average distance in large metropolitan areas was only about 12% less than in L.A."

The pressure for telecommuting will go up as fuel prices increase, says Nilles, who expects to see telecommuting first become common in Los Angeles, Chicago and New York.

The workers' advantages are



obvious, but what's in it for management? Among the 2,000 insurance company employees, Nilles noted a general increase in job satisfaction when the company decentralized and established three regional offices. Once the workers' commutes decreased, the company's turnover rate of 30% dropped sharply. "People were happier, there was less grumbling, and productivity increased 15%," Nilles reports.

Based on the insurance company study and other research, Nilles believes that telecommuting, upon first consideration, appeals most to the worker and least to management. He says managers are often nervous about telecommuting — partly because of ingrained ideas about workers and their work habits. The manager thinks, "If I can't see this guy working, how do I know he's doing anything?"

But Nilles insists that effective monitors can prevent telecommuters from "goofing off." The solution at regional offices, he says, is to have two managerial types.

"One is a traditional manager, physically located where the workers are. His function is to be sure workers are there on time and to supervise other physical aspects of the job.

"Job content and performance, on the other hand, are monitored by a specialist-manager — an accountant, a clerical supervisor or whoever is appropriate. He or she can be located anywhere, connected to the other workers by a telecommunications network."

For home workers, Nilles suggests, the simplest and most effective monitor would be to judge the quality and quantity of work at periodic intervals.

Another managerial obstacle is the "built-in" aversion many executives have toward computer technology. "The typical manager deosn't like computers — wouldn't be caught dead touching the keyboard," Nilles says. Other corporate executives have what Paul Gray, a member of Nilles's research team, calls the edifice complex. "They think they have to have a building with the company's name on it," Nilles explains.

Nilles hopes to persuade victims of the edifice complex and other nonbelievers that telecommuting can be good for business. He is seeking additional funding to study telecommuting as an alternative work pattern. "A lot of psychological barriers will have to be overcome," he acknowledges. "And some effects of telecommuting, such as possible feelings of isolation for home workers, need to be examined."

Nilles began to explore the concept of telecommuting about 10 years ago, when his work as an aerospace executive meant frequent business trips and lengthy commutes to work.

"I spent a lot of time on airplanes to Washington," he recalls. "After about 15 years of that, it got very tiresome."

These days, Nilles divides his hours between working at his USC headquarters and telecommuting from his home. His book, "Your Personal Computer: Friend or Foe?" is scheduled for publication next year.

Floppy Disk Enters Smithsonian

One of the first floppy disk drives made by Shugart Associates has become part of the Smithsonian Institution's permanent collection.

The device, donated at the National Museum of History and Technology, was the prototype of Shugart's original eight-inch single-sided floppy disk drive, the precursor of the company's SA800 series.

Introduced as the first IBM-compatible drive, the gift to the Smithsonian was the sixth floppy drive manufactured by Shugart after the company opened its doors in 1973. To date, the Sunnyvale, Calif., firm has delivered well over one-half million of these eight-inch floppys, said the company.

The prototype drive was presen-

"I agree your kit did contain a lot of RAM, and so, I can't claim fraud . . .but if you guys don't send me the interface to hook this sucker up . . .you're in big trouble!"

ted to the museum by Shugart President James Campbell, and by Don Wartner and Al Chou, two members of the original Shugart engineering team, who were instrumental in the development of floppy disk technology in the early 1970s.

According to Campbell, the acceptance by the Smithsonian of the prototype floppy drive signals recognition of one of the data processing industry's landmark product developments.

"During the 1970s the floppy disk became the standard of data interchange for word processing and small computers," Campbell said. "Today it can be found in the computing and word processing equipment that is a part of nearly every modern office. The low-cost, highly versatile floppy disk represents the presence of computer power at desk tops and work stations throughout business, government, educational institutions — even in the home."

When on exhibition, Shugart's prototype floppy disk drive may be viewed by the general public; it will also be available for study to interested visitors by appointment.

Faster Than The Reading Eye

Even after more than a century of research on the process of reading, scientists still don't understand how you read this sentence. In fact, just as aeronautical engineers once jokingly declared it theoretically impossible for a bumblebee to fly, so reading experts declare that reading is "theoretically" impossible. This is because normal literate humans read at rates far faster than their eyes could possibly comprehend individual letters or words. They can zip through sentences in a fraction of a second, sorting through a massive vocabulary to recognize words and understand their meaning.

Working as a visiting associate with Caltech Professor of Biology and Applied Science Derek H. Fender, Dr. Raymond Briggs and his colleagues William Rosar and Dr. Dennis Hocevar are making significant progress in understanding the still-mysterious process of reading.

They are in the final stages of developing a computerized machine, that is literally quicker than the eye. It is capable of changing the text on its display screen with such lightning speed that researchers can at last tease apart the complex strategies the eye and the brain use in recognizing words and coding their meaning.

The problem of inadequate reading theory is more than academic; reading is second only to speech in determining whether a person can successfully communicate in society. And despite society's massive educational apparatus, a hard core of 10 to 20 percent of people remain functionally illiterate, and a similar percentage have difficulty reading a daily newspaper. What's more, according to reading researchers, past theories have not led to successful efforts to remedy reading problems.

"The consequence of such failures for society — unemployment, poverty, delinquency, crime — read like a horror story," says reading expert Briggs. "Although this failure is partly related to long-established socio-economic factors, it is long known that reading difficulty occurs throughout the population — often in children of exceptional intelligence, from wealthy families."

Since 1879, it has been known that the eye does not fixate on each word in reading, but skitters across a sentence in a series of jumps, or saccades — about six per second in an accomplished reader. Each saccade lands several words farther along in a sentence, and so sophisticated is the mental processing that the brain even anticipates the end of a line and whips the eye back to the beginning of the next. Perception of words occurs between saccades, but the actual mental coding of what has been read may occur during the saccades themselves.

Exactly how the eye recognizes words between saccades is unknown, but several theories have been advanced. The currently most popular theory, called orthographic coding, holds that readers selectively recognize high-probability clusters and look for distictive features of words, rather than trying to figure out all the letters of all the words. The reader assumes the existence of some letters and words, rather than actually reading them, reducing the information processing required to read. Thus, you probably did not notice the missing "n" in "distinctive" several lines above.

The Caltech scientists' aim is to manipulate text so rapidly that they can precisely control what the eye is able to perceive, and can thus separate the perception of words from the cognitive processing of printed material.

To do this, they are developing a computerized "traveling window" system, drawing on pioneering research at M.I.T. Subjects tested on their system are fitted with a contact lens from which protrudes a tiny stalk with a light on the end of it. As the subject's eye moves, the beam from this light moves across a photomultiplier tube, and the information on eyeball position is fed directly into the computer. This computer also controls a display screen that can alter almost instantaneously the text the eye is reading and can record data on how the eye reacts.

For instance, the screen can be programmed to allow the subjects to see only a limited amount of text, no matter where their eye moves. Using this method, the scientists can determine how much information readers get via their peripheral vision in cognition.

Also, the device can fool the eye by altering the text in the instant before, during, or after a

saccade, to see how well the eye can reprogram its jumps and what the brain actually understands at different points in the reading process.

One major technical problem with such studies in the past, including the M.I.T. studies, has been developing a screen that can be altered in the few thousandths of a second necessary to beat the eye movement. Ordinary television-type screens cannot do this precisely because their pictures are constantly renewed by scanning, which reacts imprecisely and with some delay to commands to change displays.

The Caltech group uses a "plasma screen" as their display medium. Consisting of an array of 512 by 512 luminous dots, each of which the computer can turn on or off individually, it can change 20 or 30 characters per millisecond. The scientists are also developing high-speed computer programs fast enough to detect the beginning of a saccade and change the screen. Besides the text to be read, the screen also features two "eye buttons" in the two lower corners of the display. The computer registers glances at one or the other of the eye buttons, enabling the subjects to respond to questions about the text without taking their attention from the screen.

The researchers have been gradually working their system up to the necessary speed, also doing preliminary experiments to build up the conceptual model of the reading process they will explore using the final device. They expect to have the full system operating next year.

"Our current model accounts for about 40 percent of what's going on in reading," says Dr. Briggs, "which is better than the 15 percent we began with."

Using such technically advanced systems to study adults, says Dr. Briggs, scientists may at last come to understand how the halting, letter-by-letter reading of children is transformed into the rapid, sophisticated information processing of adults. "The future may see computerized tests, given in schools or at home to diagnose perceptual problems in children even before they begin reading. Such early diagnosis is vital to solving reading problems before the child's internal misprogramming permanently leads him astray, says Dr. Briggs.

The research is sponsored by the National Eye Institute.

New York Arson Force Fights Arson-for-Profit

New York's Mayor's Office Arson Strike Force unit has launched an information center designed to arm arson investigators with instant access to data residing in the city's departments of Finance and Housing Prevention and Development.

The Arson-for-Profit information center, a pilot project funded by the U.S. Law Enforcement Assistance Administration, is a test of computers that link together existing administrative systems to detect arson-for-profit patterns, and to identify possible suspects.

Two Raytheon Data Systems PTS-2000 information display units and a shared character printer located in Strike Force offices are connected to IBM 370 host computers in the two city agencies.

Using the systems, arson investigators and prosecutors from the city police department, the fire marshal's office, and five district attorney's offices can contact the information center, staffed nine-to-five daily, about a suspicious fire. Through the computerized data retrieval system, users will receive information regarding a building's ownership, tax history, record of arrears, classification and assessed valuation.

The computer in the Department of Housing Prevention and Development will provide information on the number of units in the building, the manager, outstanding building violations, emergency repair records, and the date of the last inspection.

Raytheon's PTS-2000 systems replace a relatively costly and time consuming manual research program. "With the manual system, vital information often was not available until just before a trial," explained Carole Sayle, director of research and analysis for the city's Strike Force. "Terminal hook-ups will allow us to routinely complete research at an

early stage of the investigative process, and also to help investigators establish leads and patterns that will help prevent torchings," Ms. Sayle continued.

There were 7,754 reported cases of arson in the city last year. Officials are hopeful that with accurate, up-to-the-minute information now at their fingertips, the computerized center will help lower that figure.

Teachers Trained in CAI

Elementary and high school students all over Bergen County, New Jersey, will soon be using microcomputers to solve mathematical equations, to compose music and find books in the school library, as a result of a new course their teachers are taking.

The workshop course, designed for Bergen teachers without previous computer experience, is being given at the Teaneck campus of Fairleigh Dickenson University.

The course was developed by Professor Stephen M. Gittleson as a series of accelerated workshop sessions to develop the teacher's competence in creating individualized instruction programs through the use of the microcomputer. Working with Silent Partner, a Fort Lee, N.J. dealer, Dr. Gittleson set up a microcomputer laboratory consisting of five Apple IIs, plus video screens, an Apple printer and software.

The course consists of four full-day sessions once a week, plus a minimum of 15 hours in the computer laboratory for each student. The summer sessions have been attended by elementary and secondary school teachers from Bergen County, who instruct pupils in a variety of subjects. Within four weeks, the teachers attending Dr. Gittleson's course were able to create programs in mathematics, music, literature, library science, history and social sciences.

For example, one librarian developed a program from which high school pupils could find all the available books in the school library on the American Presidents, from Washington right through Carter. Subjects appear on the computer's television screen, enabling the pupil to find biographies, political histories or some facet of a President's career.

Silent Partner's participation in creating the microcomputer laboratory includes extended warranties that assure complete backup and servicing of the equipment installed at the Fairleigh Dickinson campus. Silent Partner personnel have also collaborated with Dr. Gittleson and his classes in developing more advanced programs for microcomputer-aided instruction in the classroom.

☆☆☆Announcements☆☆☆

Two APL Newsletters Merge

Two APL newsletters have completed an editorial merger, according to Raymond C. Jordan, editor of *APL Market Newsletter. Personal APL News,* written by Mokurai Cherlin, contained descriptions of APL products and approaches for the personal computing user. It will now appear as a regular column in *APL Market Newsletter.* Jordan said he welcomes Cherlin's expertise in the small systems area.

APL Market Newsletter is published quarterly by Southwater Corp., a marketing and consulting company serving the APL community. Current subscription rates are $8 annually for addresses in the U.S., Canada and Mexico, and $12 for addresses elsewhere. A $2 charge is added if billing is required. For further information contact *APL Market Newsletter,* Southwater Corp., 2348 Whitney Ave., Mt. Carmel, CT 06518.

Bulletin Board

Maude (Milwaukee Area Utility for Digital Exchanges) is a computerized bulletin board which is open to the public. Messages and notices can be posted and retrieved over the telephone using a standard computer terminal or terminal emulator. The system is oriented to the needs and interests of electronics and computer hobbyists. Typical message subjects include: "Need help With...," "For Sale," "Wanted," "Have Solution for Problem With..."

A computer terminal equipped with a standard (Bell 103 compatible) modem is required to access Maude at (414) 241-8364. After about 4 rings, the modem and type carriage returns until Maude responds with a greeting. Maude will then provide additional instructions for use. The system supports 110, 300, 450 and 600 baud and operates with no ("marking") parity. When Maude is first accessed the system provides full duplex (i.e., incoming characters are echoed) and no nulls after carriage returns. Once logged on, users can change the baud, duplex mode, and number of nulls as needed.

Maude is available 24 hours/day, 7 days a week. If the system is in use, a busy signal will be received.

Maude is connected to a Thiensville, Wisconsin telephone exchange, a northern suburb of Milwaukee. Calls are free if made from anywhere within the Milwaukee Metroplan calling area. Calls from outside of the Metroplan area are billed by the telephone company as toll calls to Thiensville.

Maude exists as an experiment

in electronic communications. Suggestions for improvements or modifications are welcomed, and will be implemented as appropriate. The system is a service of a group of Milwaukee area computer hobbyists. For more information contact John Taylor, Box 121, Hartland, WI 53029.

New Newsletter

A new monthly newsletter, *Microcomputers in Education,* began publication in October. The newsletter, published by the publisher of Queue, will carry reviews of educational software, new product announcements, reports on CAI in the classrooms, reviews of books and magazine articles, news of meetings and industry news. Yearly subscriptions are $15. For information contact Microcomputers in Education, 5 Chapel Hill Drive, Fairfield, CT 06432.

OSI Users Group

The Ohio Scientific Michigan User's Group, in operation since May, has an initial membership of 130 people from primarily the southeastern Michigan area. If you're interested in joining or would like more information, contact Ralph V. Johnson, Sr., OSI-MUG, 3247 Lakewood Ave., Ann Arbor, MI; (313) 761-5358.

Workshop Services

Business and industry can now match their production requirements with vocational workshops' capabilities through a new computerized inventory program offered by the National Association of Rehabilitation Facilities.

"Hundreds of businesses and industries have found rehabilitation workshops to be reliable and cost effective," Robert C. Earnest, director of the Association's Industrial Network Services claims.

Earnest said responses to a Network questionnaire by over 500 vocational rehabilitation facilities and workshops in the United States formed the basis of the new computer inventory program.

"The survey revealed that 78 percent of the workshops are currently and primarily involved in packaging operations; 70 percent, in electrical and mechanical assembly; and 68 percent in mailing operations," he said.

"Many workshops also provide services such as electronic and electrical assembly, metal and machine work, data processing, plastics fabrication, hand and machine packaging, assembly, mailing operations and printing services," he added.

Network acts as a search coordinator between industry and workshops for the handicapped. It provides an information service to industry, and assists workshops in the development of new business contacts. To initiate the process, a business can engage the services of Network without cost or obligation. Thereafter, contracts and specific arrangements are made directly between the company and workshops with the assistance of Network.

For more information contact Robert C. Earnest, Industrial Network Services, National Association of Rehabilitation Facilities, 5530 Wisconsin Avenue, N.W., Washington, D.C., 20015; (301) 654-5882.

CP/M Software Library

The CP/M Users Group in New York has released its library of software on diskettes formatted for the North Star Horizon and North Star Micro Disk System.

Software in this library is now available on a format readable by users of double density CP/M version 1.4, double density CP/M version 2.2 and quad capacity CP/M version 2.2.

The 42 volumes in the exchange library, previously available only on 8" IBM single density CP/M format, have now been transferred volume by volume to one or two diskettes per volume.

A catalog of the software, for users of both the 8" single density and North Star formats, is available from the CP/M Users Group for $6 in the U.S., Canada and Mexico and $11 abroad. Orders must be prepared in U.S. currency.

The software is available for a media and handling charge of $8 per volume on 8" diskette, $8 or $12 per volume on North Star format (depending upon whether one or two diskettes are required) in the U.S., Canada and Mexico. Overseas users may obtain the volumes for $12 and $16 respectively. Orders must be prepaid in U.S. currency. For more information contact CP/M Users Group, 1651 Third Avenue, New York, NY 10028.

Syntax ZX80 Newsletter

Syntax ZX80, a new monthly newsletter for Sinclair ZX80 users, will begin publication this fall. Devoted to news and reviews of ZX80 hardware and software, the newsletter focuses on the new Z80A-based personal computer from Sinclair Research, Ltd., Cambridge, England.

Syntax ZX80 will also provide readers with forecasts of hardware and software, applications, and technical details for do-it-yourselfers, as well as a forum for users to share advice about programs and vendors.

Syntax ZX80 is published by The Harvard Group, an independent collection of consultants and writers. The yearly subscription rate (12 issues) is $25. Subscriptions are available by writing Ann Zevnik, Editor, The Harvard Group, Bolton Road, R.D. 2, Box 457, Harvard, MA 01451.

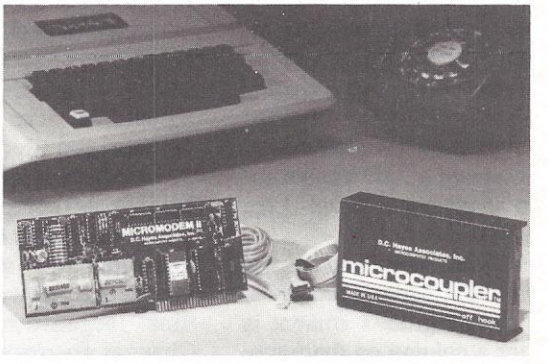
Unusual Application?

If you use your computer for an interesting application, why not write up a short (500 or 1000 words) article telling us about it? Send your submission to Random Access, *Personal Computing,* 1050 Commonwealth Ave., Boston, MA 02215.

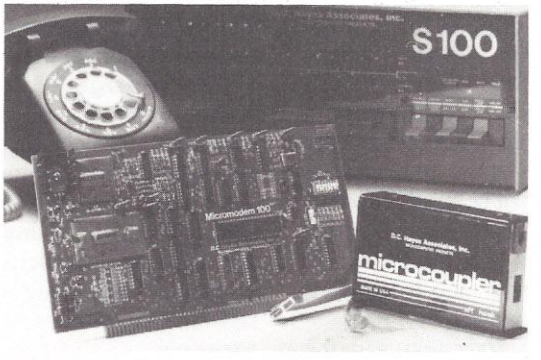
WE'D LIKE TO SHARE A FEW WORDS WITH YOU...

modem / mō′ dəm / n: A device for transmission of digital information via an analog channel such as a telephone circuit.

Micromodem II* / mī′ krō•mō′ dəm tü /*trademark* — a complete data communications system for the Apple II** Personal Computer, combining functions which formerly required a modem, an automatic calling unit, and serial and parallel interfaces. Onboard ROM firmware provides for remote console, terminal mode, and simplified implementation of more sophisticated applications with BASIC programs. The Micromodem II comes with the FCC registered Microcoupler, operates at 110 or 300 baud (Bell 103 compatible), and can automatically dial or answer the phone and transfer data.



Micromodem 100* / mī′ krō•mō′ dəm wun hun′ drəd / *trademark* — a complete data communications system for S-100 microcomputers, providing all the capabilities of a serial interface card and an acoustic coupler, with the addition of programmable automatic dialing and answer. The Micromodem 100 comes with the Microcoupler and is fully S-100 bus compatible including 16-bit machines and 4 MHz processors. The Micromodem 100 operates at either of two software selectable baud rates — 300 baud and a jumper selectable speed from 45 to 300 baud.



acoustic coupler / ə•küs′ tik kup′ lər / n: A modem that works through the standard telephone handset, transmitting data through the regular earphone and microphone. It can be affected by room noise and suffers from the distortion inherent in the carbon microphone.

Microcoupler* / mī′ krō•kup′ lər / *trademark* — an FCC registered device that provides direct access to the telephone system without the losses or distortions associated with acoustic couplers and without a telephone company supplied data access arrangement.

---

* **Micromodem II, Micromodem 100,** and **Microcoupler** are trademarks of D.C. Hayes Associates, Inc.
** Registered trademark of Apple Computer. Inc.

Hayes Microcomputer Products Inc.
5835 Peachtree Corners East, Norcross, Georgia 30092 (404) 449-8791

CIRCLE 10

The Future of Productivity in America

BY WILLIAM R. PARKS

A June special TV report, "NBC White Paper - If Japan Can...Why Can't We?," highlighted the problem of slow growth in American productivity. It was an excellent program which compared the successful Japanese industry with the declining productive economy in America. The main points centered on Japan's increased use of robots, computerization and a little known approach to management called "the statistical method."

Because of the tie-in with computers in solving some of America's problems, a review of the program's content is appropriate for a column on the future of computing. The special report was so well produced and convincing in its arguments that, as a national media event, it is sure to have some sort of impact on the future course of national policy. If it doesn't, our country will be the lesser for not having heeded the messages spelled out loud and clear.

One of the more startling findings is, in America, 85% of productivity problems are caused by management; yet in a survey conducted among managers, they blamed all the problems on workers. In Japan the statistical method involves qualitative approaches to managing people by listening to their suggestions. Because of Japan's almost universal sense of listening to workers suggestions, productivity has gone way up. A worker's ideas generally increase productivity by implementing automation while guaranteeing his job in the process. In this context a new computer system doesn't replace the worker — it frees him to do other jobs for the company to further increase the company's productivity.

The Japanese also use statistics to learn why certain aspects of productivity are down. They survey the workers to find out what's wrong. In a past column I stated something like this in the context of creating efficient new computer systems. Systems analysts should always consult the future users of their designed systems before, during and after implementing the new system. Feedback, however negative, should always be welcome by system designers and management alike.

As we enter the age of the "third" revolution — utilizing the mind machine or computer to do much of the work we still do manually — it becomes imperative that we also use the statistical method of sampling and surveying workers and users in order to increase productivity through the use of robotic devices and computer systems.

Even in the world of small computers it is important to apply such techniques. Productive software is tested ahead of time. Negative feedback is welcome. Changes are made to increase efficiency. Quality is also important. There is no room for dictating from above.

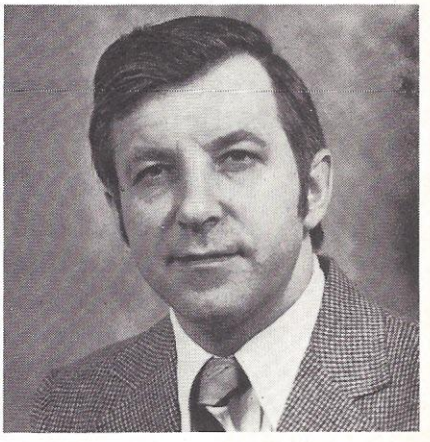
The so called "know-it-all" expert has a lot to learn. There really is no perfect expert, with all the answers, living on earth. Perhaps a little humility is in order during difficult times of declining productivity. What is intrinsically good about the Japanese system is that self-criticism is easily tolerated because job security is not tied to it. In America, there is often the opposite experience. You can be fired for pointing out your short-comings so you cover up mistakes that might lead to lowering productivity and the vicious cycle is created downward.

Increased computerization and the use of robots has not caused increased unemployment in Japan where the unemployment rate is 2%. There is job security even if the worker can find a way to automate his job with robotic devices and eliminate his present work experience. This gem of understanding has eluded many American enterprises while gaining a strong hold on the Japanese industries. The hard part of this development is that, according to the NBC report, all these ideas came from America during the period after World War II when Japan was rebuilding its industry. Americans were hired as consultants and developed this plan of statistical method, surveys and feedback. They seized on these good ideas which form the basis of the Japanese economic system. Somewhere along the line, we forgot about them in America.

America must wake up. These American-borrowed Japanese principles could gain strength once again here. As the TV report showed, there are already a number of industries implementing the statistical method of worker feedback, automation, computerization and job security.

The data processing industry will benefit as well. Data processing professionals have one of the highest turnover rates in American industry — averaging about 2 years per job. This dissatisfaction among EDP professionals can be alleviated, according to knowledgeable reports, by permitting workers greater input and recognition for the tasks they perform. At the same time, management will have to listen to the needs of their workers. Of course, the EDP workers have to do likewise with their customers in the work-place.

The message of the NBC White Paper report is really for everybody. Increased productivity through computerization and robotics is the way to go without the necessary risk of high unemployment which anti-computerists are so quick to predict. Job security can go hand-in-hand with automation if the right principles are applied in the work experience.



Professor Parks is in the Department of Mathematics and Computer Science at Elon College, Elon, NC.

MODEL II

26-4002
64K 1 Drive
$3499.00

MODEL III

26-1061 4K I.....$630.00
26-1062 16K III.....900.00
26-1063 32K III
2-Drives, RS232.....2246.00

AUTHORIZED TRS-80® DEALER A301

COMPUTER SPECIALISTS

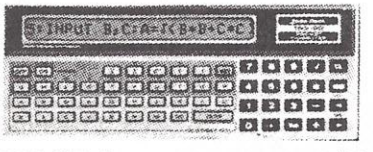
26-1056 16K Level II System with Keypad. $670.00
26-1145 RS-232 Board.....84.00
26-1140 "O" K Interface.....249.00
26-1141 "16" K Interface.....365.00
26-1142 "32" K Interface.....476.00
26-1160 Mini Disk - Drive O.....424.00
26-1161 Mini Disk - Additional.....424.00
26-1154 Lineprinter II.....720.00
26-1156 Lineprinter III.....1799.00
26-1180 Voice Synthesiser.....339.00
26-1181 VOXBOX.....145.00
26-1104 Factory Upper/Lower Case Modifcation Installed.....70.00
26-1506 Scripsit - Tape.....60.00
26-1563 Scripsit - Disk.....85.00

NOTE: Call for availability of VIDEO TEX, Model III, Color, and other new products.

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES DEDUCT 10% FROM CATALOG PRICE

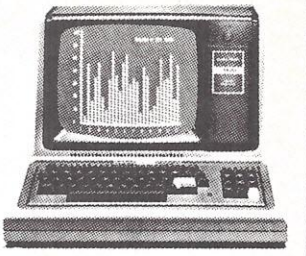
Novation Cat Modem..$149.00
CCA Data Management System.....72.00
Adventure Games Games 1-9 each.....14.00

Pocket Computer



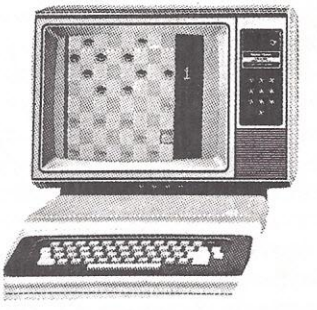
26-3501 1.9K P.C.....$225.00
26-3503 Cassette I/F.....45.00
14-812 Recorder.....72.00

MODEL I



26-1054
4K Level II
$552.00

COLOR



26-3001 4K.....$360.00
26-3002 16K.....540.00
26-3010 Color Video.....360.00
26-1206 Recorder.....54.00
26-3008 Joysticks.....22.50

Acorn Software Products, Inc.

GAMES:
Alien Invasion.....$9.00
Stock Market.....9.00
Star Trek.....9.00
Block 'Em.....9.00
Ting-Tong.....9.00
UTILITIES:
System Savers.....14.00
EDUCATION:
Language Teacher.....18.00

FREE: PRICE LIST UPON REQUEST

CENTRONICS

Fast 100 CPS Centronics
730 Printer.....$675.00
Text Quality Centronics
737 Printer.....$850.00

Model II Cobol Compiler $360.00
Cobol Run Time Package $36.00

1-800-841-0860 Toll Free Order Entry

MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of State Shipments

Immediate Shipment From Stock on Most Items

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

Full Factory Warranty on All Items Sold.

Largest Inventory In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp.

CIRCLE 11

HOLIDAY BUYER'S GUIDE

BY KEN MAZUR

Selecting a microcomputer-related gift is easier than you might think. If you know a computer buff, choosing a gift that fits the computer enthusiast's interest will be rewarding for both of you. A well-chosen holiday gift will continue to bring pleasure long after annual decorations have returned to a niche in the attic, closet of basement. An ideal gift not only supports an interest but expands enjoyment in that area as well.

The first thing to do is determine whether a computer-related gift is appropriate for the person you have in mind. Clues to help you do that will probably be as obvious as the faces on Mount Rushmore.

If a member of your family or circle of friends has been talking about personal computers or has purchased a microcomputer system, you may have noticed some behavioral traits emerging. These traits, common to the computer experience, can range from sparkling enthusiasm (with a person talking about or showing off a new system to every visitor) to downright obsession (many hours at the keyboard, total lack of time awareness, mumbling, strange symbols known as programming scrawled on every conceivable blank area and alternating periods of joy and rage over things you don't understand).

Buying a computer-related gift for persons at the enthusiastic end of the scale is easy because they're always willing to talk about anything that relates to microcomputers. In fact, you've probably noticed the biggest problem with this type of person is getting them to stop talking.

The best way to find out what the person really wants is simply to ask. Have the future recipient make a list of all the items he or she would like. Have the list put in descending order of desirability. Choosing a gift from the list is bound to make the recipient happy and still allows enough uncertainty to make opening each gift exciting.

Purchasing for persons at the obsessive end of the interest scale can be tougher because it is often hard to get their attentions away from the microcomputers long enough to find out what they want.

Gift selection for a person who rarely leaves the keyboard, or for anyone else if you insist on total surprise, will involve some work on your part but the results should be worth the effort.

To buy properly you should determine several things: the items already owned (to include the type of computer and the attachments it has, programs and books); the gift recipient's area of interest; the money you want to spend; and the level of expertise of your computer enthusiast.

Of these four items, learning the type of computer will be the most critical and the easiest. You may already know the type of machine you have in the house. If you don't, you will find the name displayed prominently on the front of the system. The difference in brands is important because programs and equipment sold for an Apple, for instance, will not run on a Radio Shack TRS-80 and vice versa.

A computer enthusiast's level of expertise can guide you in picking the right gift.

The amount of memory a computer system has is also important; programs are designed for specific machines with specific amounts of memory. A program requiring 32K of RAM and one disk drive will not run on a 16K machine with a cassette recorder. To be sure you have the right information, ask. The response you get will probably be something like "16K of RAM" or just "16K." The number (most likely 4, 16, 32 or 48) followed by a "K" tells how much random access memory (RAM) the system contains.

To be on the safe side, you should also find out what language the machine uses. An Apple computer could have Integer Basic in it or Applesoft (in fact, it could have both); a TRS-80 could be Level I Basic or Level II. There's a big difference.

After you compile the name of the computer, its memory size and language, keep this information for reference whenever you talk to computer store personnel or include it when you order from a mail order house.

You should also determine if a system uses a cassette recorder or disk drive for storing programs. Some programs are meant for one type of system or the other; but even disk programs are often distributed on cassette tape. Read the label of any considered program carefully to be sure it meets the requirements of the machine you have. Programs are usually labeled (if not right on the cassette or diskette, on the box or container that it comes in) as to machine, memory size and language if relevant. If you don't see this information, ask.

Area of interest is an excellent indication of things you can buy safely. If your enthusiast uses only prepackaged, ready-to-run programs, high-powered utility software designed to make programming easier will not be as valuable as a program intended to do a

certain job (balancing a checkbook, keeping a list of names, etc.). If the interest is computer music, shop around for something related to music rather than getting a statistical analysis program.

That leaves "level of expertise" to determine. Look at the letter to Santa. The categories of expertise are divided into three levels. In reality, the dividing lines are not nearly as exact, but the breakdown should give you a rough idea of your computer buff's expertise level.

A novice will be considered as falling into a range of persons from those just becoming interested in the microcomputer field to those starting to program in Basic. System owners in this category probably have cassette recorder-based units.

The intermediate class extends from persons beginning to develop their own applications programs to those who are mastering disk file usage. Systems would have one or more disk drives and possibly a modem and/or printer.

Advanced users are people familiar with both Basic's random access file structure and the more sophisticated features of a disk operating system, and who are dabbling with or are in command of assembly language programming. A user might also be experimenting with languages other than Basic. Systems in this category are upper-end units (full memory and such things as upper and lower case modifications, etc.) with any number of peripherals attached.

The letter has another grouping, gift type, which has been broken down into four areas: hardware (computer machinery), software (computer programs), printware (books and magazines) and accessories (odds and ends).

Within these areas is a further subdivision of price ranges that vary with the nature of the gift. An inexpensive piece of hardware costs far more than a cheap accessory. You will save a lot of time not investigating categories that are more than you want to spend. On the other hand, do not overlook the possibility of pooling resources with someone else to afford one major surprise rather than a number of small ones.

As generalizations, these are the breakdowns of prices that we consider in the various categories. In hardware, inexpensive are items costing less than $700, moderate are items between $700 and $2000 and expensive is anything over $2000. With software the ranges shift drastically lower with in-

For the Computerist:

How to Drop Hints

Wouldn't you rather get a computer-related gift during the holiday season than suffer through another bout of the necktie/scarf/school-clothes syndrome? You bet. One law of microcomputers states, "No computer system is ever complete: there's always one more item you need to be satisfied."

To ensure that you receive that special present, you can appeal to both the emotions and logic of those likely to buy you a gift in the coming weeks.

Realize that in your campaign you must use the minimum force necessary to accomplish your goal. You don't want to scar a loved one emotionally and eliminate a possible well-trained gift source in future years.

First, read through the accompanying article and circle those items you would love to find under the tree or in a sock (preferably when you're not in it). Be casual in your approach — use a wide, red felt-top marker. Leave *Personal Computing* around the house where everyone is sure to see it. Inside the refrigerator is ideal.

Next, sigh deeply while you sit at your keyboard. Talk to yourself whenever a family member passes by, saying things like, "If only I had a _____, I could _____." (Fill in your own blanks.) An occasional soul-wrenching moan might be thrown in for good measure.

If your obvious agony at not having a _____ fails to elicit expressions of concern, tune into a television soap opera to study effective methods of expressing pain verbally and facially. Two afternoons worth of study and you should be able to make even J.R. weep. Be warned: This tactic may be hazardous to your mental health. (After I tried this once, it took me a Space Invaders, two Sargon games and *three* Scott Adams' Adventures to return to a semblance of sanity.)

If the subtle approach doesn't work, threaten to go jogging without your classy running shoes. The embarrassment your family will suffer at this gauche act should be enough to make them see the light. If that doesn't work, threaten to throw yourself under the feet of some other jogger. Remind them of what a person whose body is covered with little suction cup marks looks like. If you have to carry through with the threat, you can always hire yourself out as an anti-skid bathtub mat.

As the time of gift-giving nears, step up your campaign. Make a list of items you'd like. Use the latest in communication technology to get your point across. Spray paint a suggestion list in two-foot high letters on a four-by-eight sheet of plywood. Nail the list to the bathroom wall. A captive audience is an attentive audience.

If all else fails, tell your family that lack of the items you want will force you to spend even more time at the keyboard. Be careful here: if your loved ones believe that threat, the present you find under the tree may be a divorce summons, your returned ring or an invitation to attend reform school.

If a logical approach is more to your liking, here are some phrases that have worked for others:

1. "You'll always know where I am."
2. "If I spend more time at the keyboard we won't spend so much for gas"
3. "A disk drive isn't *that* expensive: It only costs as much as 200 six-packs; or 417 Big Macs; or 1429 Mounds bars; or . . ."
4. "For you, Transcendental Meditation; for me, keyboard time."
5. "Buying me a _____ will help the economy." Or (if it's a foreign product): "Buying me a _____ will improve international relations."
6. "It's quieter than disco music."

Of course, you could always just state the truest and best reason:

7. "With encouragement and understanding I'll be able to turn out some nifty programs that will enrich all our lives."

— by Ken Mazur

expensive gifts priced at less than $15, moderates at $15 to $50 and expensive will be over $50. Accessories range from inexpensive at less than $20, through moderate, $20 and $50, to expensive for the over $50 items.

As the opening song to a current television program says, "different strokes for different folks." This buying guide is not the "end-all and be-all" of possibilities. There are literally thousands of items available in the microcomputer field that will enhance a user's experience. To cover all of them would take volumes. The possibilities outlined may not satisfy or meet the requirements of that special person of yours but they should provide you with enough information to get you started on the way to a fine holiday season. Items mentioned specifically are those which seem to have met the needs of large numbers of computer enthusiasts in the past, and, taken in that light, are fairly safe bets if you don't have much experience in the field.

The novice class of users is both easy and hard to buy for; easy because their enthusiasm makes almost any gift welcome, hard because extra care should be exercised to promote sprouting knowledge.

Hardware (machinery of some sort) is the most difficult subdivision to buy within safely as getting the proper equipment depends on knowing something about the field. If you insist on upgrading a micrcomputer, however, more memory is often a good idea. An axiom in personal computerdom is, "You never have enough memory." Computer dealers sell kits to add memory to a micro and for an added charge (maybe you can "make a deal" on this) will install that memory. Added memory (usually 16K blocks) costs anywhere from $50 to over $200. Shop around before buying. (And be sure your friend's computer doesn't already have maximum memory.)

Alternate gifts could be peripherals such as a second cassette recorder for those systems capable of handling two recorders. Cassette recorders can be purchased from $30 to $100. Two recorders provide easier storage of programs and information than just one. Before you invest in a second recorder, however, you might want to consider stretching your budget for an alternate storage (fancy way of saying "saving") device. Two such machines are the Exatron Stringy Floppy and the Beta-1 by Meca. If you can't afford one of these, look into units often referred to as high-speed tape units. These small gadgets increase the loading speeds of standard cassette recorders. The units don't give you any additional storage area but they make slow cassette recorders easier to live with.

Buying an entire personal computer system for a novice is risky; you may or may not get the proper machine for the interests of that person. Before you spend the kind of money a full system costs, you better decide that the recipient should be in on the buying process. All the various micros available have advantages and disadvantages. Characteristics you might not think are important in a machine could make the difference between joy and frustration for the owner. Don't buy a system for somebody else; let them do it. The vendor's guide accompanying this article has a list of microcomputer manufacturers.

If you're interested in seeing some of the systems listed in the vendor's guide, locate the address of a nearby computer store and visit it. Chances are excellent that the store will have a variety of machines on display. A benefit to buying locally rather than directly from a manufacturer is that your dealer will be close by if something goes wrong with the system. The manufacturers, however, will be glad to provide you with promotional literature outlining their systems, peripherals and prepackaged software.

Buying software for a novice can be immensely rewarding and good deals can be found in even the inexpensive category.

A number of computer manufacturers sell courses in programming on their systems in which the software itself helps you learn about the capabilities of the machine. If you would rather help build a library of sophisticated games (games are still one of the more popular features of a personal computer) look for something that isn't going to be boring after the first few sessions. Game subjects that maintain a challenge include Backgammon, Chess, Go, Othello, some of the more advanced space games (watch out — this class of programs ranges from "ho hum" to "holy mackerel!") and Adventure.

In the expensive category (but well worth the money) is something like Personal Software's application program VisiCalc. VisiCalc might be slightly sophisticated for a novice but it will be hard to find a better software investment. Be sure to ask about it at your local computer store.

Armed with a few basic facts, you can select a computer-related gift that will bring hours of enjoyment to your favorite personal computer enthusiast.

Inexpensive books are scarce these days but Adam Osborne's *Running Wild — The Next Industrial Revolution* is a good buy at $4 and would make an ideal stocking stuffer. You may or may not agree with what the author forsees but the small volume is bound to make you think about the future. On a less prophetic level, a series of eight books ($3.70 each or $29.60 for the series) by Russ Walter called *The Secret Guide to Computers* is excellent. Walter's style may not be to everyone's liking (sexual innuendoes appear in the strangest places) but the series is one of the most concise educations you can get in the field. Possibly the finest single volume to date on learning Basic programming with microcomputer is Dwyer and Critchfield's *Basic and the Personal Computer*. The hefty volume (438 pages) is the epitome of clear, well-organized instruction. A valuable reference guide to a novice trying to translate a published program from one version of Basic to another is Dr. David Lien's *The Basic Handbook*. Other books that will benefit a novice (at least those who have TRS-80 computers) are *Learning Level II*, also by David Lien; and *TRS-80: A Self-Teaching Guide* by Albrecht, Inman and Zamora.

A subscription to *Personal Computing* magazine has a number of advantages as a gift. Articles in *Personal Computing* spark the imaginations of readers, provide a large number of ready-to-key-in programs spanning a wide range of applications, and give the latest information in a field where a month-old product is practically out of date. *Personal Computing* offers a

continuing source of knowledge at a very low cost.

Accessories offer a multitude of gift possibilities.

Included in this category are such things as a box of blank cassette tapes, cassette tape holders (they protect tapes not in use), dust covers for the computer and its peripherals, membership in a local computer club (your local computer store can help you out on this), video display color panels that make the video green (obviously not for use with color computers), and bulk erasers (these handy gadgets allow you to erase cassette tapes and diskettes rapidly and effectively). You can also get demagnetizers for cassette recorders, which often enhance the recorder's performance.

In the intermediate price range, isolator strips that protect equipment from power surge and multiple power outlets are handly and a good buy. (Many computer systems have several electrical plugs to power the various components from wall outlets. Unfortunately, most homes only provide two outlets per wall, which can create a maze of extension wires or a rat's nest of "octopus" plugs. Neither situation is particularly desirable or safe. So a multiple power outlet can solve the problem. Simply insert the single plug into a convenient wall outlet, and then plug the various computer components into the multiple power outlet strip. The better devices have a circuit breaker to prevent electrical mishaps and a switch to turn on or off all the power outlets at once.)

For intermediate users, hardware purchases become slightly easier because user interests have already started showing up. Music synthesizer boards or add-ons are readily available for many brands of computers as are light pens. Modems, devices that allow a microcomputer to communicate over a phone line, are relatively cheap but hold tremendous potential for a computer buff. As always, more memory will be appreciated if a system isn't up to capacity yet. Voice synthesizers and voice recognition devices might be good for persons interested in verbal computer communication.

If a system is still using a cassette recorder for program storage, probably the best single gift you could buy would be a disk drive. Unfortunately, with the wide number of different drives available and the spread of price ranges, you'd best let the computerist pick out a specific system.

At this stage of development a

CUT ALONG DOTTED LINE

Dear Santa,

I am fascinated by personal computers and the potentials they offer in many areas of my life. I have filled out the enclosed guide to help you in selecting the ideal gift for me. As usual, you will find the traditional cookies and milk on the CRT. Thanks.

Yours Truly,

_____
(Name)

I consider myself to be at a ☐ novice ☐ intermediate ☐ advanced level of expertise in personal computer experience.

My equipment consists of or I want (system):

☐ Altair	☐ Commodore Pet
☐ Apple II	☐ Radio Shack TRS-80 Model I
☐ Apple II Plus	☐ Radio Shack TRS-80 Model II
☐ Apple III	☐ Radio Shack TRS-80 Model III
☐ Bally Arcade	☐ Radio Shack TRS-80 Color Computer
☐ Compucolor II	☐ Radio Shack TRS-80 Pocket Computer
☐ Digital Group	☐ Processor Tech Sol-20
☐ Exidy Sorcerer	☐ Sinclair ZX80
☐ Heath/Zenith H8	☐ SWTP 68/2
☐ Heath/Zenith H11	☐ Technical Design Labs Xitan
☐ Imsai 8080	☐ Texas Instruments 99/4
☐ North Star Horizon	☐ Vector Graphic Vector I
☐ Ohio Scientific Challenger	☐ Cromemco Z2

☐ Other _____
(Specify brand and model)

My system has ☐ 4K ☐ 8K ☐ 16K ☐ 32K ☐ 48K ☐ 64K

☐ Other _____ of RAM memory.

I use a ☐ cassette recorder ☐ 5-1/4" disk ☐ 8" disk ☐ high speed tape

☐ Other _____ for program storage.
(Specify)

The language I program in is ☐ Basic ☐ Cobol ☐ Fortran ☐ Forth ☐ Pascal ☐ Other _____.
(Specify)

My specific version of this language is _____
(Specify fully. Examples: Applesoft in RAM, Apple Integer Basic, Radio Shack Level II Basic, etc.).

My disk system uses the _____
(Specify DOS if applicable)
operating system.

(continued)

Gift areas I would like to have you consider are:

☐ Hardware

☐ full system (see above) ☐ cassette recorder
☐ 5-1/4″ disk ☐ 8″ disk
☐ memory ☐ high speed tape
☐ voice synthesizer ☐ voice recognition device
☐ programmer's calculator ☐ modem
☐ printer ☐ music synthesizer
☐ Other _____.
(Specify)

☐ Software:

☐ Applications programs in ☐ Home/Personal ☐ Business/ Finance ☐ Games ☐ Education such as _____

_____

(List the name of a particular program or general program type such as checkbook, database manager, mailing list maintainer, etc.)

☐ Utility software such as ☐ editor/assembler ☐ debugging aid

☐ programming aids such as _____

_____

(List the name of a particular program or the type of utility you want such as a line renumberer, fast sort routine, machine language modules, etc.)

☐ A programming course for my particular machine.

☐ A disk operating system such as _____
(TRSDOS, NEWDOS, SOS 3.3, CP/M, etc.)

☐ A tie-in to a computer communication network.

☐ Programming language package: ☐ Compiler Basic ☐ Cobol

☐ Fortran ☐ Pascal ☐ Forth ☐ Other _____
(Specify)

☐ Printware

☐ A subscription to *Personal Computing*.

☐ The book titled _____.

☐ Any book dealing with _____.
(Fill in your area of interest)

☐ Accessories

I think the following accessories would be nice to have :

☐ blank cassette tapes ☐ cassette tape holders
☐ dust covers ☐ computer club membership
☐ video display panels ☐ bulk erasers
☐ recorder head demagnetizer ☐ isolator strip
☐ multiple power outlet ☐ blank diskettes
☐ copy holder ☐ floppy disk saver kit
☐ printer paper ☐ diskette protectors
☐ furniture ☐ carrying case
☐ Other _____
(Specify)

Use the space below to list other gift ideas or to expand on particulars of items checked above.

_____

_____

_____

_____

_____

CUT ALONG DOTTED LINE

printer will be valuable, too, but the same situation holds true as for the disk drive; let the person to use the machine pick it out.

Software is going to start in the moderate price range for intermediate users programming their own computers. "Utility" software that helps a person program more effectively is available for most personal computers and can be a blessing to someone who wants to program a machine faster and more easily. Have your intended recipient thumb through some computer magazines and circle the programs that might help him or her.

If a system already has a modem or there is talk of getting one, check into the consumer-oriented, microcomputer communication networks offered by The Source and CompuServe Information Service. (The September issue of *Personal Computing* provides details on these systems.)

Disk Operating Systems provide good value for the money if you know what disk system is on a machine; and a flexible database management system will prove its worth fairly rapidly. If you're going to spend a lot of money on software, be sure you get the right programs for the computer your friend has. Check with a local computer store or read computer magazines before you buy.

Finding a book valuable for an intermediately skilled computer user won't be easy almost volumes around are either designed for a beginner or an engineer. For the TRS-80 owner with disk drives, one of the most valuable is Harvard C. Pennington's *TRS-80 Disk and Other Mysteries*. There's more information packed in this volume for a Radio Shack disk-based machine than you'll find anywhere else. Beyond that, much of what you'll find are general books on database management. Be warned, however, that most database management volumes read like college textbooks.

Accessories for this group of users include blank diskettes; copy holders (they hold material from which a program is being keyed into the machine); floppy disk saver kits (reinforce the center of heavily used diskettes); storage boxes or vinyl sleeves to protect diskettes while not in use; paper for a printer; furniture (desks for a microcomputer designed for that specific machine and printer stands); and carrying cases (to protect a system if you take it from one place to another).

Advanced users will also be hard to buy for because there's a lot of material

— by Ken Mazur

available in the field that just won't interest them. If the person is getting into assembly language programming, a programming calculator that converts number systems will be handy. Texas Instruments puts one out called the "TI Programmer" for about $60.

Beyond that, hardware buys will be tricky because an advanced user is more likely to want advanced equipment such as 8-inch disk drives (instead of the 5 1/4-inch models) or hard disks for increased speed and storage. They'll also probably want a printer with some pretty fancy options, as opposed to a novice or intermediate level user who might be glad to get any printer at all. A gift certificate, while not totally satisfying emotionally, could be your best option and may save a lot of frustration.

If you decide to buy software, look into those programs that will help an assembly language programmer or which offer increased microcomputer capabilities to those people who know how to use them. Editor/Assemblers aren't overly expensive and are available for a number of computers.

Assembly language programming might not be the "cup of tea" for even an advanced user but there are a number of high-level language packages that could be interesting. Languages like Fortran, Cobol and Pascal have been designed to run on microcomputers. Also available are packages for CP/M (an operating system from Digital Research that offers many advantages to persons wanting to use a microcomputer in a business application) and compiler versions of Basic.

Assembly language books abound, with some aimed at general principles and others dealing with particular microprocessors. For a TRS-80 owner just starting assembly language, REMsoft's REMASSEM-1 combination of audio cassette - microcomputer assisted course featuring William Barden's *TRS-80 Assembly Language Programming* as a text will be helpful. The course retails for $70.

Accessories can be any of those mentioned for the less advanced users.

Most vendors are honest, reliable businesspeople who go out of their way to keep a customer satisfied. Unfortunately, there are always exceptions. You are more likely to be satisfied with the buyer-seller relationship if, before actually purchasing the gift of your choice, you observe some basic cautions.

Be sure you know exactly what you're getting. Ask specifically what is included in the package you're buying. On some pieces of equipment, for instance, you may have to buy special cables to attach the unit to the microcomputer. If the cables aren't included, you will have to pay extra for them and that might increase the price more than you want to pay. In other cases, the cables will be included so you can have the unit up and running with no additional expense. Double check to be sure everything you're supposed to be getting is actually in the box before you leave the computer store.

Visiting a local computer store allows you to see, first hand, the hundreds of potential gift items available.

If a telephone number is given in the vendor's guide call before you order to see if the item you want is in stock. Some vendors offer items that they in turn have to order from someone else. If all you have is an address but a vendor seems to be the only one offering the item you want, maintain a correspondence file and photostat every piece of correspondence you mail to the vendor; should problems occur, you will have an accurate record of what transpired. An added caution that will cost little in the long run is sending all correspondence certified, registered or return receipt requested mail as you may want proof that your order arrived at the vendor's.

Keep all receipts! The purchase of software sometimes comes with future updates included in the price (or available at a reduced price for purchasers of a previous version of that program). Rules on updates vary, so it is a good idea to ask specifically what your friend will need to prove ownership of a gift that you buy. Tell the vendor what you're doing and find out the best way to handle the purchase.

Also check on return policies. Some software is easy to copy and the vendor may not allow a program to be returned once it's purchased. The best thing to do is discuss the return policy with the vendor before you buy software.

When you purchase by mail order, it's often wise to pay by credit card. If problems arise between you and the vendor, using your credit card gives you added time to stop payment.

And finally, a last recommendation: whenever you have a question about the microcomputer field, *ask*. Your local computer store is an excellent place to get answers, especially if you are seeking information that could result in a purchase. The best method for getting answers is to visit the store; it gives you the opportunity to see a variety of products that you might not have originally considered and allows store personnel to show you the items they talk about. Calling a store is an alternative; you may get juggled from one phone extension to another for a while but generally you'll find someone with the knowledge and willingness to help if you endure. Given the economic condition of the country, vendors are falling all over themselves trying to sell products and maintain a healthy cash flow — take advantage of the trend. Also, call or visit more than one place since people have an understandable tendency to be biased toward the products they sell. Check around before you spend your money.

With the basic approaches and knowledge outlined, you should be able to buy a computer-related gift with success. If you're still unsure of your ability to get the right thing, arrange to visit a computer store with your friend and watch to see what gets him or her excited. You can revisit the store by yourself later to further investigate the items pointed out in the first visit.

There is one final warning. You are liable to catch the computer enthusiasm bug once you begin to consider personal computers as something other than machines you don't understand. I'm sure you don't fully understand every process and part that makes your car run but not knowing the specifics doesn't mean you can't drive effectively. The computer mystique is an imaginary barrier that anyone can break down. If you find yourself becoming enthralled with the expanding world of microcomputers you could easily become the computer enthusiast *your* friend will buy a gift for; if that turns out to be true, read "How to Drop Hints."□

VENDOR GUIDE

Hardware

Systems

Apple Computer Inc.
10260 Bandley Dr.
Cupertino, CA 95014
(408) 996-1010
Circle 150

Atari Consumer Division
1265 Borregas Ave.
PO Box 427
Sunnyvale, CA 94086
(800) 538-8547
Circle 151

Commodore Business Machines Inc.
3330 Scott Blvd.
Santa Clara, CA 95051
(408) 727-1130
Circle 152

Compucolor Corp.
PO Box 569
Norcross, GA 30071
(404) 449-5879
Circle 153

Hewlett-Packard Co.
Corvallis Division
1000 NE Circle Blvd.
Corvallis, OR 97330
(503) 757-2000
Circle 154

Mattel Electronics
5150 Rosecrans Ave.
Hawthorne, CA 90250
(213) 644-0411
Circle 155

Ohio Scientific Inc.
1333 S. Chillicothe Rd.
Aurora, OH 44202
(216) 562-5177
Circle 156

Radio Shack
1300 One Tandy Center
Fort Worth, TX 76102
(817) 390-3011
Circle 157

Texas Instruments Inc.
Personal Computer Division
2301 N. University
Lubbock, TX 79408
(806) 741-3737
Circle 158

Other

ALF Products
1448 Estes
Denver, CO 80215
Apple II music board — $265
Circle 159

American Square Computers
Kivett Dr.
Jamestown, NC 27282
(919) 883-1105
Peripherals & systems
Circle 160

Anadex Inc.
9825 DeSoto Ave.
Chatsworth, CA 91311
(213) 998-8010
Printers
Circle 161

Berliner Computer
102 Jericho Tpke.
New Hyde Park, NY 11040
Memory kits for Apple, TRS-80
Circle 162

California Computer Systems
250 Caribbean Dr.
Sunnyvale, CA 94086
(408) 734-5811
Apple II arithmetic processor unit
Circle 163

Cameo Data Systems
1626 Clementine St.
Anaheim, CA 92802
(714) 535-1682
Cartridge controller for TRS-80 Model II
Circle 164

Electronics Specialists Inc.
171 South Main St.
Natick, MA 01760
(617) 655-1532
Isolator strips: $57-$97
Circle 165

Exatron
3555 Ryder St.
Santa Clara, CA 95051
(408) 737-7111
Stringy Floppy
Circle 166

JPC Products Co.
12021 Paisano Ct.
Albuquerque, NM 87112
(505) 294-4623
High speed cassette system: $120
Circle 167

Lobo Drives
935 Camino Del Sur
Goleta, CA 93017
(805) 685-4546
Disk drives for Apple, TRS-80, S-100
Circle 168

Matchless Systems
18444 S. Broadway
Gardena, CA 90248
(213) 327-1010
Peripherals for Apple, TRS-80, Pet
Circle 169

Meca
70260 O.W.S. Rd.
Yucca Valley, CA 92284
(714) 365-7686
Beta-1: $400
Circle 170

MicroMint Inc.
917 Midway
Woodmere, NY 11598
(516) 374-6793
Remote control unit for TRS-80, Apple, S-100
Circle 171

Microtek Inc.
9514 Chesapeake Dr.
San Diego, CA 92123
(714) 278-0633
Memory expansion for TRS-80
Circle 172

Mountain Hardware
300 Harvey West Blvd.
Santa Cruz, CA 95060
(408) 429-8600
Peripherals for Apple
Circle 173

Netronics R&D Ltd.
333 Litchfield Rd.
New Milford, CT 06776
(800) 243-9375
In CT (203) 354-9375
Computer terminal kit
Circle 174

Parasitic Engineering
Box 6314
Albany, CA 94706
(415) 839-2636
8″ disks for TRS-80: $995
Circle 175

Simutek
PO Box 13687-Z
Tucson, AZ 85732
(602) 886-5880
Peripherals for TRS-80
Circle 176

Vista Computer Co.
Dept PI
Torrance, CA 90503
(213) 320-2880
Disk Systems
Circle 177

Software

Applications

CalData Systems
PO Box 178446
San Diego, CA 92117
(714) 272-2661
Word processing for TRS-80 Mod II
Circle 178

Computer Systems Design Inc.
PO Box 735
Yakima, WA 98907
(509) 575-0320
Business software for TRS-80, Pet, IBM
Circle 179

Creative Discount Software
256 S. Robertson
Suite 2156
Beverly Hills, CA 90211
(800) 824-7888
Applications for Apple, TRS-80
Circle 180

Data Train Inc.
840 NW 6th St.
Suite 3
Grants Pass, OR 97526
(503) 476-1467
Payroll program for TRS-80: $235
Circle 181

Edu-Ware Services Inc.
22035 Burbank Blvd.
Suite 223
Woodland Hills, CA 91367
(213) 346-6783
Educational Software for Apple
Circle 182

Hayden Book Co.
50 Essex St.
Rochelle Park, NJ 07662
(201) 843-0550
Applications, games for Apple, TRS-80
Circle 183

Information Unlimited Software Inc.
281 Arlington Ave.
Berkeley, CA 94707
(415) 525-9452
Word processing for Apple II
Circle 184

Level IV Products
32238 Schoolcraft, #F4
Livonia, MI 48154
(313) 525-6200
Software for TRS-80
Circle 185

Microsoft Consumer Products
10800 Northeast Eighth
Suite 507
Bellevue, WA 98004
General applications
Circle 186

RTR Software Inc.
PO Box 12351, Dept A6
El Paso, TX 79912
(915) 544-4397
Applications for Apple II
Circle 187

The Bottom Shelf
PO Box 49104
Atlanta, GA 30359
(404) 939-6031
Applications & utilities
Circle 188

TYC Software
40 Stuyvesant Manor
Geneseo, NY 14454
(716) 243-3005
Educational software for Apple, TRS-80
Circle 189

Games

Adventure International
Box 3435
Longwood, FL 32750
(305) 862-6917
Adventure games for TRS-80, Apple II, Apple II Plus, Sorcerer
Circle 190

Barclay Bridge
8 Bush Ave.
Port Chester, NY 10573
(914) 937-4200
Bridge on Apple II
Circle 191

Automated Simulations
PO Box 4232
Mountain View, CA 94040
(800) 824-7888
In CA (800) 852-7777
Dunjonquest series of games
Circle 192

Computer Cablevision
2617 42d St., NW
Washington, DC 20007
Gammon Challenger for TRS-80
Circle 193

Dynacomp
PO Box 162 Dept P
Webster, NY 14580
Flight simulator & Bridge 2.0
Circle 194

Peter Frey
2407 Prospect Ave.
Evanston, IL 60201
Othello for TRS-80, Apple
Circle 195

Palmer, McBride & Kincaid Assoc.
PO Box 598
East Brunswick, NJ 08816
(201) 246-7680
Chess: Sargon, Boris 2.5
Circle 196

Programma International Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579
Apple invader
Circle 197

Strategic Simulations Inc.
Dept PC
450 San Antonio Rd.
Suite 62
Palo Alto, CA 94306
(800) 648-5600
Games for TRS-80, Apple
Circle 198

Quality Software
6660 Reseda Blvd.
Suite 103
Reseda, CA 91335
(213) 344-6599
Fastgammon
Circle 199

Utilities

BPS
322 West 57th St.
New York, NY 10019
(212) 765-0815
High speed sort for OSI
Circle 200

Datacope
PO Box 55053
Hillcrest Station
Little Rock, AR 72205
Disk sort for Apples
Circle 201

Pickles & Trout
PO Box 1206
Goleta, CA 93017
(805) 967-9563
CP/M for TRS-80 Model II
Circle 202

Racet Computes
702 Palmdale
Orange, CA 92665
(714) 637-5016
Utilities for TRS-80
Circle 203

Web Associates
PO Box 60
Monrovia, CA 91016
(714) 559-6249
Utilities for TRS-80
Circle 204

Courses

Program Design Inc
11 Idar Court
Greenwich, CT 06830
(203) 661-8799
"Step-by-Step"
Basic programming course for TRS-80
Circle 205

REMsoft
571 East 185
Euclid, OH 44119
(216) 531-1338
"REMASSEM-1"
Assembly language course for TRS-80
Circle 206

General

Aardvark Technical Services
1690 Bolton
Walled Lake, MI 48088
(313) 624-6316
Games, utilities for OSI
Circle 207

Clark Systems Corp.
PO Box 490156
Atlanta, GA 30349
Software for Health systems
Circle 208

JJR Data Research
Box 74
Middle Village, NY 11379
(516) 643-1931
Games for North Star systems
Circle 209

Potters Programs
22444 Lakeland
St. Clair Shores, MI 48081
(313) 573-8000
Utilities for North Star systems
Circle 210

Programma International Inc.
3400 Wilshire Blvd.
Los Angeles, CA 90010
(213) 384-0579
Software for Apples
Circle 211

Quality Software
6660 Reseda Blvd.
Suite 103
Reseda, CA 91335
(213) 344-6599
Software for Atari, Sorcerer
Circle 212

Printware

Basic and the Personal Computer
by Dwyer and Critchfield
Addison-Wesley Publishing Co.
Reading, MA
Circle 213

Basic Handbook Learning Level II
by Dr. David Lien
Compusoft Publishing
1050 E. Pioneer Way
El Cajon, CA 92020
(714) 588-0996
Circle 214

Chess (variety of titles)
David McKay Co. Inc.
750 Third Ave.
New York, NY 10017
Circle 215

H&E Computronics magazine
50 N. Pascack Rd.
Spring Valley, NY 10977
(914) 425-1535
Written for TRS-80 owners
Circle 216

Instructo/McGraw-Hill
Cedar Hollow Rd.
Paoli, PA 19301
Instructo paper computer
Circle 217

Personal Computing magazine
1050 Commonwealth Ave.
Boston, MA 02215
(617) 232-5470
Circle 218

Running Wild — The Next Industrial Revolution
by Adam Osborne
Osborne/McGraw Hill Inc.
630 Bancroft Way
Berkeley, CA 94710
Circle 219

The Secret Guide to Computers
by Russell M. Walter
92 Saint Botolph St.
Boston, MA 02116
Circle 220

TRS-80 Disk & Other Mysteries
by Harvard Pennington
IJG Computer Services
569 N. Mountain Ave.
Suite B
Upland, CA 91786
Circle 221

Accessories

Advanced Access Group
10526 W. Cermak
Westchester, IL 60153
(312) 562-5210
Diskette protection box
Circle 222

Alpha Supply Co.
9625 Mason Ave.
Chatsworth, CA 91311
(213) 882-9818
Diskettes
Circle 223

Audio-Video Systems
2485 Autumval Ave.
San Jose, CA 95165
(408) 946-1265
Furniture for TRS-80
Circle 224

Bits Inc.
PO Box 428
25 Route 101 West
Peterborough, NH 03458
(800) 258-5477
In NH (603) 924-3355
Programming pads for assembly language programmers
Circle 225

Computer Plus+ Inc.
15 Mare Lane
New Milford, CT 06776
(203) 354-0167
Desks & printer stands
Circle 226

Computer TexTile Inc.
10960 Wilshire Blvd.
Suite 1504
Los Angeles, CA 90024
(213) 477-2196
Apple carrying case
Circle 227

Cover Craft
PO Box 555
Amherst, NH 03031
(603) 889-6811
Dust covers for microcomputers
Circle 228

Devoke Data Products
3780 Fabian Way
Palo Alto, CA 94303
(415) 494-8844
Furniture for TRS-80
Circle 229

Lifeboat Associates
2248 Broadway
New York, NY 10024
(212) 580-0082
Head cleaning diskettes
Circle 230

National Tricor Inc.
3335 Greenleaf Blvd.
Kalamazoo, MI 49008
(616) 375-7519
Green video display for TRS-80
Circle 231

Novation Inc.
18664 Oxnard St.
Tarzana, CA 91356
(800) 423-5410
In CA (213) 996-5060
Modems
Circle 232

3M Data Recording Products
Dept DR80-1
Box 33600
St. Paul, MN 55133
(612) 733-9572
Head cleaning diskettes
Circle 233

Tri-Star Corp.
PO Box 1727
Grand Junction, CO 81501
(303) 243-5200
Floppy disk saver kit
Circle 234

Wallace Computer Supplies
1024 West Wilcox
Peoria, IL 61604
(309) 685-7876
Apple accessories
Circle 235

Tracking Add-On Sales

BY CLINT HENTZ

I operate several sales and service departments in a mulit-store operation. It became a way of life to know if the sales or service staff was selling a specific "add-on" or accessory item they were instructed to push. In addition, it's important to know what the relationship is between the various stores in this area of operation.

For example, in the area of reupholstering some stores also make a clear protective plastic slip cover to be put over the new upholstery fabric. While the customer's furniture is in the shop for reupholstery work, it's profitable for the shop to make the protective cover, and it's advantageous to have the cover sold in connection with a reupholstery order.

The program comes into play when tracking the productivity of the various salespersons and the individual stores in this type of operation. Sending the results from the program to the respective store managers provides them with the statistics to motivate low producers. With your copy you can see if the store managers are effective in stimulating their people.

The print-out produced by the listed program seems to be well accepted by my staff; they feel they were judged fairly and accurately. The program could also track the salesperson's productivity in the following areas: Service contracts — TV and major appliances; Protective finishes — carpet cleaning; Detergent — automatic washers; Training courses — microcomputers.

A second feature of the program will produce a telephone soliciation form for making future sales. It contains sufficient pertinent information which can be incorporated into the phone pitch. There are places next to each customer's name to record the date and time the call was made, and also a place to record the dispositon of the call. This information becomes useful when substantiating the productivity of the solicitation crew.

This program was written on a 48K TRS-80 with FD200 Pertec disk drives, TRSDOS 2.2 and a Radio Shack tractor feed line printer which is the same as a Centronics 779. The program takes a little less than 6K; it could be utilized with ease on a smaller TRS-80, if you do not need to handle a large amount of transactions.

With TRSDOS 2.2 you could sub-

Mr. Hentz is president of a custom furniture manufacturing and reupholstering company and operations manager for a service center handling TV and major appliance repair, carpet handling and installation, furniture refinishing, drapery making and monogramming. His previous PC *articles include "Personalized Sales Message" (July 1980) and "Drapery Estimating" (September 1980).*

Telephone Solicitation Form

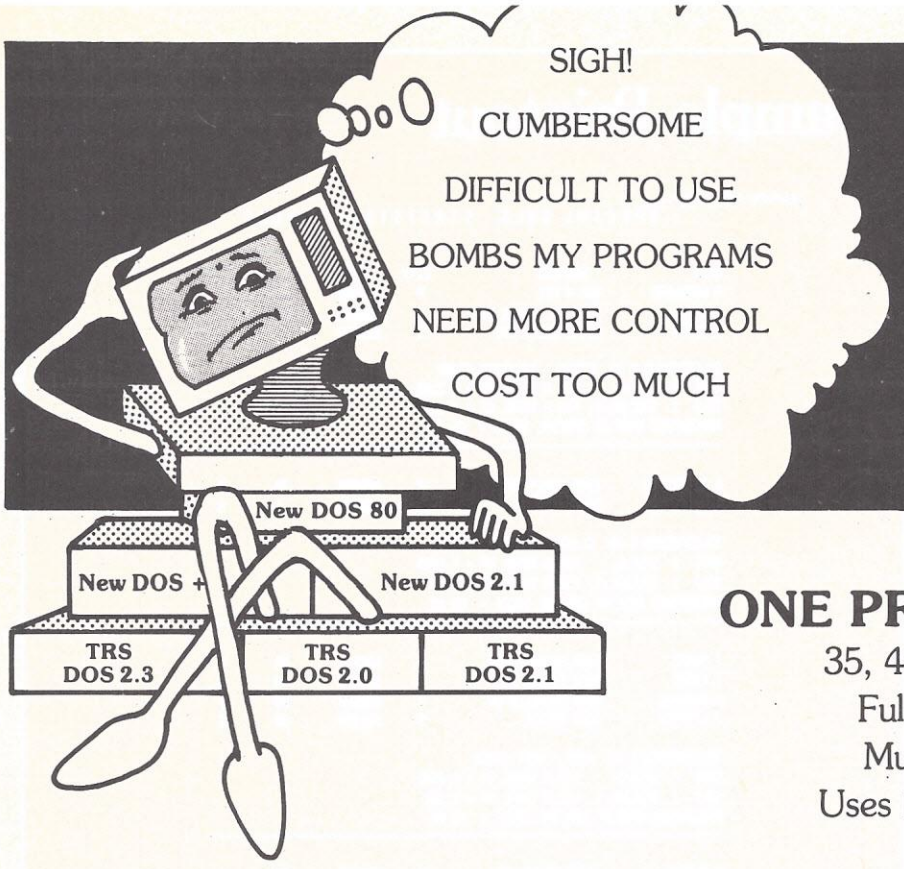
DATE	TIME	PHONE	Y/N	NAME	ADDRESS	CITY	ZIP	SALES#	CLASS	SPEC$	RETAIL$	DATE
----	----	6473734	----	J. HENTZ	1980 AMERICA AVE	S	63199	30	1	50.00	750.00	102180
----	----	4564343	----	R. BARNHARD	BOX 4 RT1	P	63399	18	0	0.00	680.00	100180
----	----	9875678	----	J. JONES	9999 ST. LOUIS AVE	S	63111	18	2	60.00	890.00	100380
----	----	7659898	----	R. IMPOSSIBLE	101234 HILL ST	B	62206	44	1	58.00	450.00	101680
----	----	6788765	----	E. JOHNSTON	5656 MISSISSIPPI AVE	S	63166	40	0	0.00	433.00	101880
----	----	4568888	----	D. ALBERTSON	1980 MARCH AVE	P	63399	20	2	78.00	675.00	102080
----	----	1239876	----	K. ALASKA	10001 DEGREE ST	S	63001	33	1	60.00	455.00	101980
----	----	6453535	----	U. BAKER	890 OVEN DR	S	63111	40	0	0.00	308.00	101080
----	----	8765678	----	A. FRANCO	6789 UNIVERSITY DR	S	63113	44	0	0.00	569.00	101980
----	----	6546787	----	W. WILLIAMS	9999 CURVE ST	B	62220	33	2	80.00	678.00	100580
----	----	9877890	----	B. BARNHARD	BOX 3 RT2	P	63303	20	0	0.00	788.00	102580
----	----	6545656	----	S. SMITH	12345 BLANK AVE	S	63130	18	1	59.00	530.00	101580
----	----	7655678	----	T. KYSER	5656 CIRCLE DR	B	62220	30	0	0.00	320.00	100580
----	----	8765432	----	S. SANDERS	8990 COCHELLA DR	S	63125	30	1	70.00	625.00	101880
----	----	6543456	----	J. LINDSEY	6666 GOETHE	S	63109	30	2	90.00	566.00	102380

THERE ARE 15 TELEPHONE NUMBERS ON THIS REPORT.

CUSTOMERS CALLED ---------- CUSTOMERS SOLD ---------- PERCENT EFFECTIVE ----------

Data Statements (Contained on separate file and merged with program.)

```
3000 DATA J. HENTZ,1980 AMERICA AVE,S,63199,6473734,30,1,50,750,102180
3010 DATA R. BARNHARD,BOX 4 RT1,P,63399,4564343,18,0,0,680,100180
3020 DATA J. JONES,9999 ST. LOUIS AVE,S,63111,9875678,18,2,60,890,100380
3030 DATA R. IMPOSSIBLE,101234 HILL ST,B,62206,7659898,44,1,58,450,101680
3040 DATA E. JOHNSTON,5656 MISSISSIPPI AVE,S,63166,6788765,40,0,0,433,101880
3050 DATA D. ALBERTSON,1980 MARCH AVE,P,63399,4568888,20,2,78,675,102080
3060 DATA K. ALASKA,10001 DEGREE ST,S,63001,1239876,33,1,60,455,101980
3070 DATA U. BAKER,890 OVEN DR,S,63111,6453535,40,0,0,308,101080
3080 DATA A. FRANCO,6789 UNIVERSITY DR,S,63113,8765678,44,0,0,569,101980
3090 DATA W. WILLIAMS,9999 CURVE ST,B,62220,6546787,33,2,80,678,100580
3100 DATA B. BARNHARD,BOX 3 RT2,P,63303,9877890,20,0,0,788,102580
3110 DATA S. SMITH,12345 BLANK AVE,S,63130,6545656,18,1,59,530,101580
3120 DATA T. KYSER,5656 CIRCLE DR,B,62220,7655678,30,0,0,320,100580
3130 DATA S. SANDERS,8990 COCHELLA DR,S,63125,8765432,30,1,70,625,101880
3140 DATA J. LINDSEY,6666 GOETHE,S,63109,6543456,30,2,90,566,102380
3150 DATA X,X,X,1,1,1,1,1,1,1
```



Perk up your TRS-80
—with—
Level IV DOS 4.1
—and—
Level IV Basic (C)
—by—
Vernon B. Hester

ONE PRICE INCLUDES:

35, 40, 77, or 80 Tracks
Full Utilities Package
Multi-Auto Function
Uses Less RAM Memory!

Compatible with all commonly used operative systems!

Get all the "Bells & Whistles
without the glitches & problems!

Only **$119.95**

Limited Time Introductory Offer

$89.95

TRS-80 & New DOS 80 are trademarks of Tandy & Apparat Co's.

LEVEL IV PRODUCTS, INC.

32238 Schoolcraft Road, Suite F4 • Livonia, MI 48154
313-525-6200 Outside Michigan call 1-800-521-3305

Please add $2.50 for shipping and handling.
COD add another $1.25 plus 25% deposit;
balance certified check or cash.

Dealers Orders Welcome
Open Tues. thru Sat. 11-7 E.S.T.

CIRCLE 13

stitue APPEND for MERGE but you need to enter two file specs with APPEND but only one with MERGE. I suggest you type in the program and save it on a diskette under a file name, such as "Sales." Prepare the data statements on a separate file such as "Mar," for the March sales or "Oct," for the October sales. Depending on the amount of data statements, you may be able to put two months on one side of a diskette. If you are using the Pertec drives you can put two more on the other side of the diskette. If your drive will not accept both sides of a diskette, you'll need six diskettes to hold the twelve months of data in place of the three with the Pertec drives. I suggest keeping the program on a diskette by itself.

To use the program, load "Sales," remove the diskette and insert the proper month's data diskette. Type in MERGE"OCT", assuming you are with the October data, and in a flash you will have the program and the October data statements together in the computer. You are then ready to run the sales statistics report or the telephone soliciation report.

You must remember to start numbering the data statements with a number which is higher than any used in the program. You will see I have used 3000, an easy number to remember. Each month's data statements should start with the number 3000. Remember, you must save everything with the A option. For example, after typing in October's data, type in SAVE"OCT", A or the MERGE will not work properly.

If you do not have a disk drive write the program, include one month's data statements in the program and CSAVE on tape. When the next month's data is ready for entry, CLOAD the previously made tape and delete all of the data statements. Type in the current months data on the existing program and then CSAVE on a separate tape. Now you have two tapes each containing a program and a given month's data. It is really a simple system, except the tape seems so slow to me.

Another reason I like the disk system I outlined is because it is very easy to use a given month of data statements with several other programs I have written, such as a mailing label program, and another which creates a personalized promotional sales letter. Of course, there are other disk programming techniques which can handle data, but the one in this article seems to work so well and it doesn't require any additional programming to alter, delete or update the data. □

Sample Printout

```
                    SPECIAL SALE  STATISTICS REPORT

NAME              ADDRESS              CITY    DATE      $ SPEC.  CLASS#
R. BARNHARD       BOX 4 RT1             P     100180       0       0
J. JONES          9999 ST. LOUIS AVE    S     100380      60       2
S. SMITH          12345 BLANK AVE       S     101580      59       1

SALESPERSON # 18  PRESS - DOWNTOWN STORE.
MADE  3  SALE/S     FOR A TOTAL OF = $ 2100
SOLD  2  SPECIAL/S. FOR A TOTAL OF = $ 119
PERCENTAGE RATE OF SPECIAL SALES =  66.7%
-------------------------------------------------------------------------------

D. ALBERTSON      1980 MARCH AVE        P     102080      78       2
B. BARNHARD       BOX 3 RT2             P     102580       0       0

SALESPERSON # 20  CLINT - DOWNTOWN STORE.
MADE  2  SALE/S     FOR A TOTAL OF = $ 1463
SOLD  1  SPECIAL/S. FOR A TOTAL OF = $ 78
PERCENTAGE RATE OF SPECIAL SALES =  50.0%
-------------------------------------------------------------------------------

J. HENTZ          1980 AMERICA AVE      S     102180      50       1
T. KYSER          5656 CIRCLE DR        B     100580       0       0
S. SANDERS        8990 COCHELLA DR      S     101880      70       1
J. LINDSEY        6666 GOETHE           S     102380      90       2

SALESPERSON # 30  JUNE - WEST STORE.
MADE  4  SALE/S     FOR A TOTAL OF = $ 2261
SOLD  3  SPECIAL/S. FOR A TOTAL OF = $ 210
PERCENTAGE RATE OF SPECIAL SALES =  75.0%
-------------------------------------------------------------------------------

K. ALASKA         10001 DEGREE ST       S     101980      60       1
W. WILLIAMS       9999 CURVE ST         B     100580      80       2

SALESPERSON # 33  RUTH - WEST STORE.
MADE  2  SALE/S     FOR A TOTAL OF = $ 1133
SOLD  2  SPECIAL/S. FOR A TOTAL OF = $ 140
PERCENTAGE RATE OF SPECIAL SALES = 100.0%
-------------------------------------------------------------------------------

E. JOHNSTON       5656 MISSISSIPPI AVE  S     101880       0       0
U. BAKER          890 OVEN DR           S     101080       0       0

SALESPERSON # 40  RODGER - EAST STORE.
MADE  2  SALE/S     FOR A TOTAL OF = $ 741
SOLD  0  SPECIAL/S. FOR A TOTAL OF = $ 0
PERCENTAGE RATE OF SPECIAL SALES =   0.0%
-------------------------------------------------------------------------------

R. IMPOSSIBLE     101234 HILL ST        B     101680      58       1
A. FRANCO         6789 UNIVERSITY DR    S     101980       0       0

SALESPERSON # 44  BETTY - EAST STORE.
MADE  2  SALE/S     FOR A TOTAL OF = $ 1019
SOLD  1  SPECIAL/S. FOR A TOTAL OF = $ 58
PERCENTAGE RATE OF SPECIAL SALES =  50.0%
-------------------------------------------------------------------------------


                         STORE  TOTALS

 DOWNTOWN    SOLD   3  OUT OF A TOTAL OF  5  FOR   60.0%
 - RETAIL OF REG. SALES = $ 3563  SPEC. = $ 197
-------------------------------------------------------------------------------


 WEST        SOLD   5  OUT OF A TOTAL OF  6  FOR   83.3%
 - RETAIL OF REG. SALES = $ 3394  SPEC. = $ 350
-------------------------------------------------------------------------------


 EAST        SOLD   1  OUT OF A TOATL OF  4  FOR   25.0%
 - RETAIL OF REG. SALES = $ 1760  SPEC. = $ 58
-------------------------------------------------------------------------------

                      TOTAL  GOAL  ALL  STORES    50.0%
                      ----------------------------------

                 AVERAGE SPECIALS SOLD ALL STORES   56.1%
                 DIFFERENCE FROM GOAL                6.1


TOTAL REGULAR SALES (ONLY)          $ 8717.00
TOTAL SPECIAL SALES (ONLY)          $  605.00
TOTAL ALL SALES                     $ 9322.00

TOTAL NUMBER OF REGULAR SALES  = 15
TOTAL NUMBER OF SPECIAL SALES  = 9
```

Program Listing on page 36

THE ORIGINAL MAGAZINE FOR OWNERS OF THE TRS-80™* MICROCOMPUTER

SOFTWARE FOR TRS-80™ OWNERS

H & E COMPUTRONICS INC.

MONTHLY NEWSMAGAZINE FOR TRS-80™ OWNERS

MONTHLY NEWSMAGAZINE
Practical Support For Model I & II

- PRACTICAL APPLICATIONS
- BUSINESS
- GAMBLING • GAMES
- EDUCATION
- PERSONAL FINANCE
- BEGINNER'S CORNER
- NEW PRODUCTS
- SOFTWARE EXCHANGE
- MARKET PLACE
- QUESTIONS AND ANSWERS
- PROGRAM PRINTOUTS

. . . . AND MORE

PROGRAMS AND ARTICLES PUBLISHED IN OUR FIRST 12 ISSUES INCLUDE THE FOLLOWING:

- A COMPLETE INCOME TAX PROGRAM (LONG AND SHORT FORM)
- INVENTORY CONTROL
- STOCK MARKET ANALYSIS
- WORD PROCESSING PROGRAM (FOR DISK OR CASSETTE)
- LOWER CASE MODIFICATION FOR YOUR VIDEO MONITOR OR PRINTER
- PAYROLL (FEDERAL TAX WITHHOLDING PROGRAM)
- EXTEND 16-DIGIT ACCURACY TO TRS-80™ FUNCTIONS (SUCH AS SQUARE ROOTS AND TRIGONOMETRIC FUNCTIONS)
- NEW DISK DRIVES FOR YOUR TRS-80™
- PRINTER OPTIONS AVAILABLE FOR YOUR TRS-80™
- A HORSE SELECTION SYSTEM***ARITHMETIC TEACHER
- COMPLETE MAILING LIST PROGRAMS (BOTH FOR DISK OR CASSETTE SEQUENTIAL AND RANDOM ACCESS)
- RANDOM SAMPLING***BAR GRAPH
- CHECKBOOK MAINTENANCE PROGRAM
- LEVEL II UPDATES***LEVEL II INDEX
- CREDIT CARD INFORMATION STORAGE FILE
- BEGINNER'S GUIDE TO MACHINE LANGUAGE AND ASSEMBLY LANGUAGE
- LINE RENUMBERING
- AND CASSETTE TIPS, PROGRAM HINTS, LATEST PRODUCTS COMING SOON (GENERAL LEDGER, ACCOUNTS PAYABLE AND RECEIVABLE, FORTRAN-80, FINANCIAL APPLICATIONS PACKAGE, PROGRAMS FOR HOMEOWNERS, MERGE TWO PROGRAMS, STATISTICAL AND MATHEMATICAL PROGRAMS (BOTH ELEMENTARY AND ADVANCED) . . . AND

FREE

WORD PROCESSING PROGRAM **(Cassette or Disk)** For writing letters, text, mailing lists, etc., with each new subscriptions or renewal.

LEVEL II RAM TEST **(Cassette or Disk)** Checks random access memory to ensure that all memory locations are working properly.

DATA MANAGEMENT SYSTEM **(Cassette or Disk)** Complete file management for your TRS-80™

CLEANUP **(Cassette or Disk)** Fast action Maze Game

ADVENTURE **(Cassette or Disk)** Adventure #0 by Scott Adams (From Adventureland International)

FREE

* TRS-80™ IS A TRADEMARK OF TANDY CORP.

SEND FOR OUR NEW 48 PAGE SOFTWARE CATALOG (INCLUDING LISTINGS OF HUNDREDS OF TRS-80™ PROGRAMS AVAILABLE ON CASSETTE AND DISKETTE). $2.00 OR **FREE** WITH EACH SUBSCRIPTIONS OR SAMPLE ISSUE.

H & E COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

master charge

24 HOUR ORDER LINE
(914) 425-1535

VISA

NEW TOLL-FREE ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

ONE YEAR SUBSCRIPTION $24 .....
TWO YEAR SUBSCRIPTION $48 .....
SAMPLE OF LATEST ISSUE $ 4 .....
START MY SUBSCRIPTION WITH ISSUE .....
(#1 - July 1978 • #7 - January 1979 • #12 - June 1979 • #18 - January 1980)
NEW SUBSCRIPTION ..... RENEWAL .....

NEW!!!
MOD-II NEWSLETTER
$12/year (or 12 issues)

CREDIT CARD NUMBER _____ EXP. DATE _____

SIGNATURE _____

NAME _____

ADDRESS _____ CITY _____ STATE _____ ZIP _____

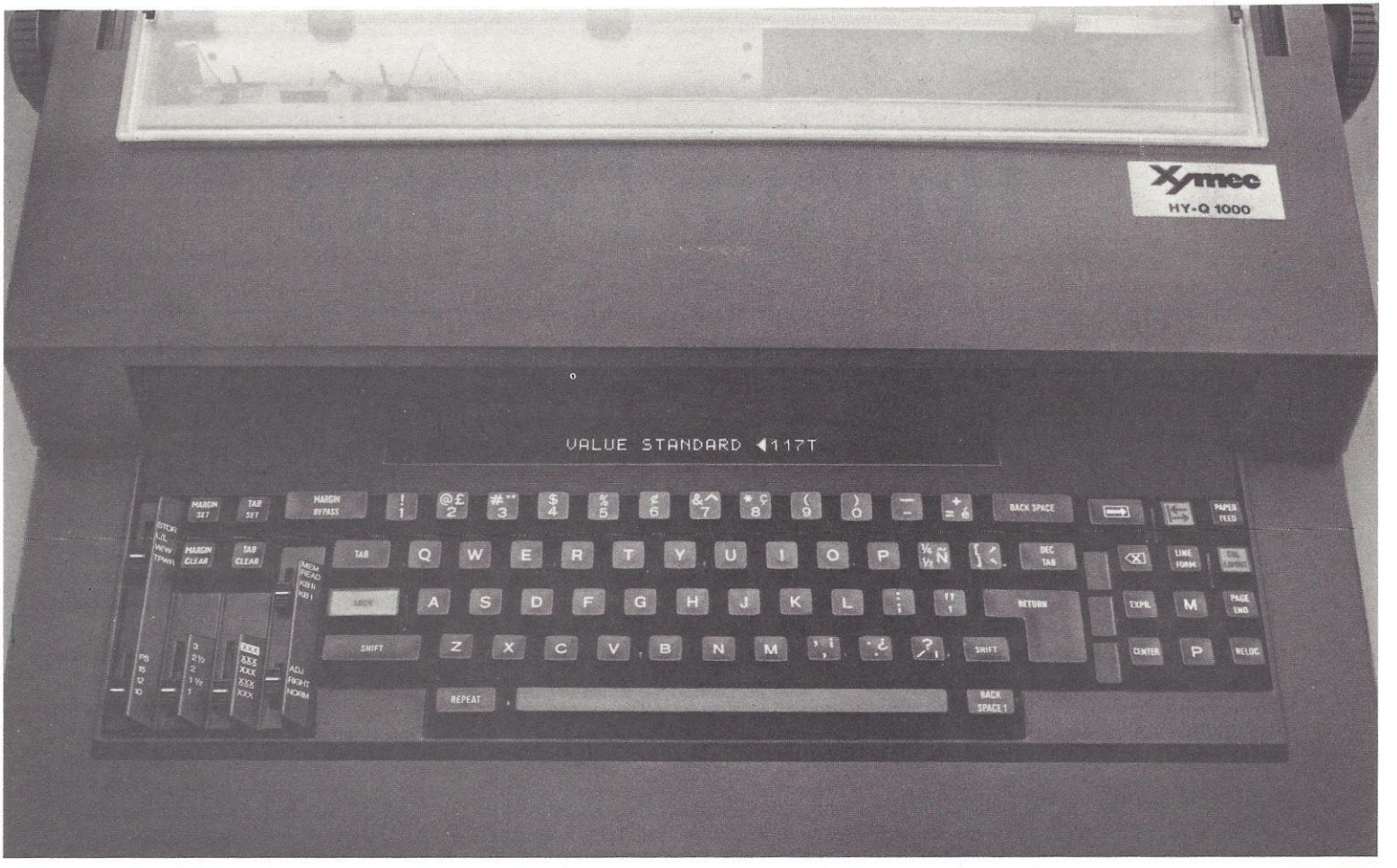
***** ADD $6/YEAR (CANADA, MEXICO) - ADD $12/YEAR AIR MAIL - OUTSIDE OF U.S.A., CANADA & MEXICO *****

CIRCLE 14

Program Listing

```
10 REM  TRACKING ADD-ON SALES & PHONE SOLICITATION FORM
20 REM         C L I N T   H E N T Z
30 REM          ST. LOUIS, MISSOURI
40 PRINT:PRINT:PRINT"DID YOU MERGE PROPER DATA FILE ?"
50 REM    THE : SEPERATES MULTI LINE STATEMENTS ON TRS-80
60 PRINT"============================="
70 REM    IF NOT ABLE TO USE MULTI LINE STATEMENTS MAKE 4 SEPERATE PRINT STATEMENTS
80 PRINT:PRINT:PRINT:PRINT
90 PRINT"                  1.  TELEPHONE  SOLICITATION  LIST"
100 PRINT"                  2.  SPECIAL ITEM STATISTICS REPORT"
110 PRINT
120 PRINT"                        ENTER SELECTION"
130 INPUTU9
140 ON U9 GOTO 150  ,380
150 LPRINT"                              T E L E P H O N E   S O L I C I T A T I O N   R E P O R T":LPRINT" "
160 REM    LPRINT SENDS INFO TO PRINTER
170 REM    LPRINT WITH DOUBLE QUOTES PRINTS BLANK LINE ON HARD COPY
180 LPRINT"DATE  TIME  PHONE  Y/N       NAME              ADDRESS";
190 LPRINT"          CITY    ZIP   SALES#  CLASS SPEC$  RETAIL$  DATE"
200 LPRINT"-------------------------------------------------";
210 LPRINT"-------------------------------------------------";
220 LPRINT"----------"
230 LPRINT"  "
240 READ A$,B$,C$,D,E#,F,G,H,I,J
250 IF A$="X" GOTO 310
260 X$="----  ---- ####### ----  %           % %               % % %  #####  ###     ## ###.## ####.##  ######"
270 LPRINT USING X$;E#,A$,B$,C$,D,F,G,H,I,J
280 REM    T6 COUNTS NUMBER OF NAMES
290 T6=T6+1
300 GOTO 240
310 LPRINT"  "
320 LPRINT TAB(33)"THERE ARE ";T6;" TELEPHONE NUMBERS ON THIS REPORT."
330 LPRINT"  "
340 LPRINTTAB(14)"CUSTOMERS  CALLED ---------- CUSTOMERS SOLD ---------- PERCENT EFFECTIVE ----------"
350 END
360 REM     A$=CUST NAME-B$=STREET-C$=CITY STATE-D=ZIP-E#=PHONE#
370 REM     F=SALES#-G=CLASS-H=SPECIAL$-I=RETAIL$-J=DATE
380 LPRINT TAB(8)" S P E C I A L  S A L E   S T A T I S T I C S  R E P O R T":LPRINT"  ":LPRINT"  "
390 LPRINT"NAME              ADDRESS              CITY     DATE      $ SPEC.  CLASS#"
400 REM      O= NUMBER OF SALES PERSONS
410 FOR O= 1 TO 6
420 REM      THE #  IS AFTER E TO PRINTOUT 7 DIGETS
430 READ A$,B$,C$,D,E#,F,G,H,I,J
440 IF A$="X" GOTO 640
450 REM     O1=SALESPERSONS ID#
460 IF O=1 THEN O1=18
470 IF O=2 THEN O1=20
480 IF O=3 THEN O1=30
490 IF O=4 THEN O1=33
500 IF O=5 THEN O1=40
510 IF O=6 THEN O1=44
520 H1=H1+1
530 IF F=O1 GOSUB600
540 REM     T(0) ADDS NO. OF REGULAR SALES
550 IF F=O1 THEN T(O)=T(O)+1
560 REM      EE(0) ADDS RETAIL $
570 IF F=O1 THEN EE(O)=EE(O)+I
580 IF (F=O1)AND(I>0 )THEN G7(O)=G7(O)+H
590 GOTO 430
600 M$="%           % %              %    % % ######     ###     ###"
610 IF G>0 THEN Y(O)=Y(O)+1
620 LPRINT USING M$;A$,B$,C$,J,H,G
630 RETURN
640 RESTORE
650 LPRINT"  "
660 E1=Y(1)+Y(2):E2=Y(3)+Y(4):E3=Y(5)+Y(6)
670 IF O1=18 THEN S$="PRESS"
680 IF O1=20 THEN S$="CLINT"
690 IF O1=30 THEN S$="JUNE"
700 IF O1=33 THEN S$="RUTH"
710 IF O1=40 THEN S$="RODGER"
720 IF O1=44 THEN S$="BETTY"
730 IF O1=18 THEN S1$="DOWNTOWN"
740 IF O1=20 THEN S1$="DOWNTOWN"
750 IF O1=30 THEN S1$="WEST"
760 IF O1=33 THEN S1$="WEST"
770 IF O1=40 THEN S1$="EAST"
780 IF O1=44 THEN S1$="EAST"
790 Y1=T(1)+T(2):Y2=T(3)+T(4):Y3=T(5)+T(6)
800 LPRINT"SALESPERSON #"O1;" ";S$;" - ";S1$;" STORE."
810 LPRINT"MADE ";T(O);" SALE/S     FOR A TOTAL OF = $";EE(O)
820 LPRINT"SOLD ";Y(O);" SPECIAL/S.  FOR A TOTAL OF = $";G7(O)
830 IF (T(O)=0)AND(Y(O)=0) THEN BB=0:GOTO 880
840 GG$="PERCENTAGE RATE OF SPECIAL SALES = ###.#%"
850 REM    DIVIDING TWO NUMBERS & MULTIPLYING BY 100 GIVES %
860 BB=Y(O)/T(O)*100
870 LPRINT USING GG$;BB
880 LPRINT"-------------------------------------------------------------------------------"
890 LPRINT"  "
900 NEXT
910 LPRINT"  ":LPRINT"  ":LPRINT"  "
920 LPRINT"                         STORE  TOTALS"
930 W4$=" ###.#%"
940 REM    950 PREVENTS DIVISION BY ZERO ERROR
950 IF E1=0 AND Y1=0 LPRINT"DOWNTOWN  NO SALES  % = 0":GOTO1010
960 LPRINT"  "
970 LPRINT " DOWNTOWN    SOLD  ";E1;" OUT OF A TOTAL OF ";Y1;" FOR ";:LPRINT USING W4$;E1/Y1*100
980 LPRINT" - RETAIL OF REG. SALES = $";EE(1)+EE(2);" SPEC. = $";G7(1)+G7(2)
990 LPRINT"-------------------------------------------------------------------------------"
1000 LPRINT"  "
1010 IF E2=0 AND Y2=0 LPRINT" WEST     NO SALES  % = 0":GOTO1070
1020 LPRINT"  "
1030 LPRINT" WEST        SOLD  ";E2;" OUT OF A TOTAL OF ";Y2;" FOR ";:LPRINT USING W4$;E2/Y2*100
1040 LPRINT" - RETAIL OF REG. SALES = $";EE(3)+EE(4);" SPEC. = $";G7(3)+G7(4)
1050 LPRINT"-------------------------------------------------------------------------------"
1060 LPRINT"  "
1070 IF E3=0 AND Y3=0 LPRINT" EAST       NO SALES  % = 0":GOTO1200
1080 LPRINT"  "
1090 LPRINT" EAST        SOLD  ";E3;" OUT OF A TOATL OF ";Y3;" FOR ";:LPRINT USING W4$;E3/Y3*100
1100 LPRINT" - RETAIL OF REG. SALES = $";EE(5)+EE(6);" SPEC. = $";G7(5)+G7(6)
1110 LPRINT"-------------------------------------------------------------------------------"
1120 REM  1130-1150 ALLOWS ANY ONE STORE TO HAVE NO SALES
1130 IF Y1=0THENE7=((E3/Y3*100)+(E2/Y2*100))/2
1140 IF Y3=0THENE7=((E1/Y1*100)+(E2/Y2*100))/2
1150 IF Y2=0THENE7=((E1/Y1*100)+(E3/Y3*100))/2
1160 IF Y1 ANDY2 ANDY3>0 THEN E7=((E1/Y1*100)+(E2/Y2*100)+(E3/Y3*100))/3
1170 KK=Y(1)+Y(2)+Y(3)+Y(4)+Y(5)+Y(6)
1180 H2=E1+E2+E3:HH=H2/H1*100
1190 LPRINT"  "
1200 SS$="               AVERAGE SPECIALS SOLD ALL STORES  ###.#%"
1210 G4=50:IF E7>G4 THEN J9=E7-G4 ELSE J9=G4-E7
1220 H6$="               DIFFERENCE FROM GOAL            ###.#"
1230 LPRINT"                       TOTAL  GOAL  ALL  STORES    50.0%"
1240 LPRINT"                       ----------------------------------":LPRINT"  "
1250 LPRINT USING SS$;E7
1260 LPRINT USING H6$;J9
1270 REM    J3= TOTAL SPECIAL SALES
1280 J3=G7(1)+G7(2)+G7(3)+G7(4)+G7(5)+G7(6)
1290 REM    J2= TOTAL REGULAR SALES
1300 J2=EE(1)+EE(2)+EE(3)+EE(4)+EE(5)+EE(6)
1310 LPRINT"  ":LPRINT"  "
1320 J4$="TOTAL REGULAR SALES (ONLY)          $#####.##"
1330 LPRINT USING J4$;J2
1340 J5$="TOTAL SPECIAL SALES (ONLY)          $#####.##"
1350 LPRINT USING J5$;J3
1360 J6$="TOTAL ALL SALES                     $#####.##"
1370 LPRINT USING J6$;J3+J2
1380 LPRINT"  "
1390 LPRINT"TOTAL NUMBER OF REGULAR SALES  =";T(1)+T(2)+T(3)+T(4)+T(5)+T(6)
1400 LPRINT"TOTAL NUMBER OF SPECIAL SALES  =";KK
1410 LPRINT"  ":LPRINT"  "
1420 LPRINT"CLASS # OF SPECIALS"
1430 LPRINT"--------------------"
1440 LPRINT"CLASS #1 = REGULAR CLEAR PLASTIC SLIP COVER"
1450 LPRINT"CLASS #2 = DELUX CLEAR PLASTIC COVER"
```

Illustrated by Josh Randall



Now... Intelligence beyond your CRT

HY-Q 1000

Xymec's HY-Q 1000™ Series brings microprocessor capability to printing. This fast, letter quality, intelligent printer/typewriter expands the capability of your computer or word processing system.

Xymec gives your system more of the features you want most:

- **L.E.D. display**
- **tractor feed**
- **RS232, IEEE-C779 interface**

Compatible with nearly all microcomputers, the HY-Q 1000™ Series prints **boldface** and reverse, centers titles, sets columns and right justifies. Your choice of three pitches and proportional spacing, with minimum throughput print speed of 20 cps. Changing print wheels or type styles takes but seconds.

Intelligent printing makes good sense to me: Send me more information and the name of my local dealer.

Name _____

Title _____

Company _____

Address _____

City _____

State _____ Zip _____

Telephone _____

Xymec, Inc.
17905 Sky Park Circle, Suite J
Irvine, CA 92714

Xymec™
The Value Standard in Printers

Xymec gives you more features than any other machine on the market, regardless of price!

Xymec intelligent printers won't create a service nightmare. Reliably constructed, the HY-Q 1000™ Series is built with serviceability in mind. And with service centers nationwide, maintenance is easy and convenient.

Give your computer system the intelligent edge. Join up with Xymec. For more information, send in this coupon today.

Xymec, Inc.
Subsidiary of Litronic Industries
17905 Sky Park Circle, Suite J
Irvine, CA 92714 (714) 557-8501

CIRCLE 15

Backup Deleter

BY DON WOOD

Good programming practice demands that you make frequent back-up copies of programs you are developing (or even copying from a printed listing). If the program is at all long or complicated, you're unlikely to finish it at one sitting, so you'll want to save your incomplete program for later. Even after you "finish" the program, you'll still have to debug it, again probably over several sessions with the computer. And once you have it debugged and running, you'll probably find that modifications, expansions and refinements suggest themselves — and once again you'll be making backup copies of different versions of the program. Your disk will quickly fill up with unusable fragments, bug-riddled programs and multiple variations on your software theme.

Backup Deleter can help keep your disks uncluttered. When all those backups are no longer useful, run this program to clean them out and make room on your disk for new projects.

The programs presented here were written in Applesoft on an Apple II Plus running DOS 3.2. However, you should be able to implement the simple, underlying concept on other systems with similar DOS capabilities.

Quite simply, this program deletes multiple backups of a given program. Sure, you could erase them manually — but that quickly gets tedious for more than two or three programs: UNLOCK FILE. Whirr, clack. DELETE FILE. Whirr, clack. UNLOCK FILE. Whirr, clack... Hence, Backup Deleter.

Naming Files

To benefit from this program (and also to impose some order on your backup files), you'll need to follow some definite pattern in naming your files. You can pick whatever pattern suits your fancy; the important thing is that the file names differ only in a sequential number used as part of the file name.

For an example, let's look at the pattern I use. Suppose I'm developing a program called TEST. I may save several incomplete versions before I finish typing the code into the computer:

TEST-1/INC
TEST-2/INC
TEST-3/INC

and so forth. The "INC", of course, stands for "Incomplete." The numbers 1, 2 and 3 refer to the first, second and third SAVEs of the incomplete program TEST.

Once typed in, TEST may require several sessions to work out all the bugs:

TEST-1/BUGS
TEST-2/BUGS
TEST-3/BUGS

and so forth. Here, "BUGS" indicates that the program isn't yet in working order; and the numbers indicate the sequence in which these bug-riddled versions were SAVEd.

Even after debugging, TEST may continue to evolve with new features, modifications, extensions and revisions. At this point, I can drop the slash (/) and suffix, and just use program name and sequential number:

TEST-1
TEST-2
TEST-3

and so forth.

Fast and Easy

A brief examination of Program Listing 1 will show how Backup Deleter works.

First, note that each file name, as constructed above, can be divided into three parts. The first part (F$ in the program) consists of all letters and other characters in the file name preceding the sequential number. The second part is the sequential number. And the last part (L$) is everything following the number. (If you devise your own scheme for naming backup files, keep this triple division in mind.)

Now look at Program Listing 1. Line 20 defines D$ as Control-D, needed later to activate disk commands. For example, line 30 uses D$ to turn on the Apple DOS's MON C,I,O feature.

MON C,I,O is not essential to the program's operation, but it is handy because it lets you see what's happening at each step of the program's operation. For example, when the program unlocks and deletes a file called TEST-1/INC, the computer prints on the screen:

UNLOCK TEST-1/INC
DELETE TEST-1/INC

This is repeated for each file the program unlocks and deletes. In other words, MON C,I,O lets you monitor the program's interaction with the disk by printing on the screen each disk command the program issues. Without MON C,I,O you'd stare at an unchanging screen until the program finished all its deletions.

If your DOS lacks a similar capability, you can simply add a few lines to the program to print out the same information:

85 PRINT "UNLOCK "; N$
95 PRINT "DELETE "; N$

For an example of Backup Deleter in operation, let's assume we want to delete five incomplete TEST programs named TEST-1/INC, TEST-2/INC and so

Listing 1 – Fast and Easy

```
 10 REM BACKUP DELETER
 20 D$ = CHR$ (4): REM CTRL-D
 30 PRINT D$; "MON C,I,O"
 40 F$ = "TEST-"
 50 L$ = "/INC"
 60 FOR J = 1 TO 5
 70 N$ = F$ + STR$ (J) + L$
 80 PRINT D$; "UNLOCK "; N$
 90 PRINT D$; "DELETE "; N$
100 NEXT J
110 PRINT: PRINT "DELETIONS COMPLETED"
```

forth. Lines 40 and 50 define F$ and L$ as the first and last parts of the file name. Thus, F$ = "TEST-" and L$ = "/INC". Note that the punctuation (the hyphen and the slash) must be included.

Lines 60 and 100 establish the loop that does the program's work. We let J go from 1 to 5 because this is the range of sequential file numbers we wish to delete.

In line 70, STR$(J) converts the loop variable J into its string equivalent. This string is concatenated with the first and last part of the file name to construct the complete file name, N$.

Lines 80 and 90 (using D$ to activate the disk commands) UNLOCK and DELETE the backup file. Line 100 loops back to 60 to continue the process for the next file. When all deletions are completed, line 110 tells us the program has finished.

Of course, the version of Backup Deleter in Program Listing 1 only works for one particular set of files. But the principle is so simple you could easily write a Backup Deleter for any files you want to erase. On the other hand, you may prefer (as I do) to keep a polished "library version" of Backup Deleter handy.

Library Version

This version, given in Program Listing 2, customizes itself to any set of files according to your inputs.

Let's take a look at Program Listing 2 (which, by the way, occupies less than 1.5K). Following the header information (lines 100 to 200), line 210 defines D$ as Control-D; as in the "fast and easy" version, D$ is used later to activate disk commands. "MON C,I,O" in line 220 lets you see on the screen each UNLOCK and DELETE sent to the disk by the loop in lines 570 to 610.

Lines 230 to 430 simply print header information and instructions. Since Backup Deleter is not a program you'll use every day, you'll probably want to keep these brief instructions as a reminder, though they play no part in the program's operation.

The VTABs and HTABs scattered throughout this section are the Apple's vertical and horizontal tab functions, used to produce a visually pleasing screen display. If you're adapting this program for another computer, substitute your own Basic's formatting features (PRINT @, TAB, etc.) to print a display that fits your particular screen size.

The input section, from 440 to 530, is straightforward. Lines 460 and 480 request the first and last parts of the file name, while 510 and 520 input the range of sequential file numbers you want deleted. Remember to include punctuation (hyphens, slashes, etc.) when entering the file name parts. If your file name has no "last" part (for example, TEST-1), simply enter Return; L$ will then be an empty string ("") and will play no further role in the program.

Also, note that the file numbers you input must be sequential with no intermediate numbers missing. If the program tries to UNLOCK a non-existant file named TEST-3, the program crashes.

Lines 540 to 560 give you a last chance to change your mind before actually deleting the files.

The loop from 560 to 610 — the heart of the program — is identical to the loop in the "fast and easy" version, except that the loop limits have been change to A and B, the numbers you input in lines 510 and 520.

Line 620 tells you the deletions are completed, and line 640 gives you the option of either reviewing the disk's CATALOG or ending the program immediately. Line 680 signals the program's end. □

Listing 2 – Library Version

```
100 REM BACKUP DELETER
110 REM BY DON WOOD
120 REM
130 REM
140 REM COPYRIGHT 1980
150 REM PERSONAL COMPUTING
160 REM 1050 COMMONWEALTH AVE.
170 REM BOSTON, MA  02215
180 REM
190 REM
200 REM
210 D$ = CHR$ (4): REM CTRL-D
220 PRINT D$; "MON C,I,O"
230 HOME
240 VTAB 3: HTAB 13: PRINT "BACKUP DELETER"
250 VTAB 4: HTAB 14: PRINT "BY DON WOOD"
260 VTAB 5: HTAB 13: PRINT "COPYRIGHT 1980"
270 VTAB 6: HTAB 11: PRINT "PERSONAL COMPUTING"
280 VTAB 10
290 PRINT "THIS PROGRAM DELETES BACKUP FILES"
300 PRINT "WHICH HAVE NAMES OF THE FORM"
310 PRINT
320 HTAB 12
330 PRINT "FFFFFFNLLLLLL"
340 PRINT
350 PRINT "WHERE 'FFFFFF' IS THE FIRST PART"
360 PRINT "OF THE FILE NAME, 'LLLLLL' IS THE"
370 PRINT "LAST PART, AND 'N' IS THE SEQUENTIAL"
380 PRINT "NUMBER OF THE FILE"
390 PRINT
400 PRINT "FILE NUMBERS MUST BE IN SEQUENTIAL"
410 PRINT "ORDER WITH NO NUMBERS MISSING"
420 PRINT
430 INPUT "PRESS 'RETURN' TO CONTINUE";I$
440 HOME
450 VTAB 8
460 INPUT "WHAT IS THE FIRST PART OF THE FILE NAME? ";F$
470 PRINT
480 INPUT "WHAT IS THE LAST PART OF THE FILE NAME? ";L$
490 PRINT
500 PRINT "NOW INPUT THE RANGE OF FILE NUMBERS YOU WANT
    DELETED."
510 INPUT "WHAT IS THE FIRST FILE NUMBER? ";A
520 INPUT "WHAT IS THE LAST FILE NUMBER?  ";B
530 PRINT
540 PRINT "PRESS 'RETURN' TO DELETE FILES"
550 INPUT "OR 'E' TO END ";I$
560 IF I$<>"" THEN 680
570 FOR J = A TO B
580 N$ = F$ + STR$ (J) + L$
590 PRINT D$; "UNLOCK "; N$
600 PRINT D$; "DELETE "; N$
610 NEXT J
620 PRINT: PRINT: PRINT: PRINT "DELETIONS COMPLETED"
630 PRINT: PRINT: PRINT
640 PRINT "PRESS 'RETURN' FOR CATALOG"
650 INPUT "OR 'E' TO END ";I$
660 IF I$<>"" THEN 680
670 PRINT D$; "CATALOG"
680 PRINT "PROGRAM FINISHED": END
```

Translating TRS-80 Level II Basic to TI 99/4 Basic

BY HARLEY M. TEMPLETON

When a new computer comes on the market, it takes time for software to be developed for it. Not everyone wants to write programs; many computer owners prefer programs ready-made. One way to provide programs for a new computer is to translate (or transport) programs written for another computer to the new one. It would be nice to have all those TRS-80 programs available for the new kid on the block, the TI 99/4. This article will help you transport TRS-80 Level II programs to your Texas Instruments computer.

First, let's review some features of the TI 99/4. It comes with a color monitor, audio tone generator and provision for two cassette recorders. It has 16K bytes of ROM, and uses a 16-bit microprocessor, the TMS 9900.

Subprograms change the color of the screen and the colors of characters to provide impressive color graphics. Audio tones add significantly to the effect of displays. Now you can write a quiz program that displays WRONG! in full color, to the tune of "No, No, A Thousand Times, No!" If you're not that ambitious, you can settle for a raucous "Baaah!" to discourage wrong answers.

How about the 16-bit microprocessor? Well, memory accesses are 16 bits at a time, and results are computed 16 bits at a time. The TI 99/4 uses eight bytes for the internal representation of a number: seven bytes for the mantissa (numeric value) and one byte for the exponent. If the exponents of two numbers were equal, adding 16 bits at a time means that four executions of the computer's add instruction would be required to add the numbers. In an 8-bit microprocessor, seven executions of the add instruction would be required for the same computation. Of course, this is a generalization, but the comparison is valid; the 16-bit microprocessor tends to be faster and more efficient because it computes a larger portion of the result with each execution of an instruction.

Single precision computations on the TI 99/4 provide 10 digits of accuracy. Altogether, 13 digits are computed, and the 10 most significant digits are displayed.

TI claims that their 99/4 Basic conforms to the American National Standard for Minimal Basic, a standard of the American National Standards Institute (ANSI). The purpose of the standard is to promote transportability; a program written in a version of Basic that conforms closely to the standard should require few changes to run on another computer that conforms to the standard. Radio Shack has never claimed that TRS-80 Basic conforms to any standard, so some changes in a Level II Basic program are required. But the transportability is good; you *can* run TRS-80 programs on the TI 99/4.

Comparing the Languages

Tables 1, 2, 3 and 4 list 63 statements and functions of TRS-80 Level II Basic

Table 1 — Program Statements

TRS-80 Level II	TI 99/4
CLEAR	Not required
CLS	CALL CLEAR
DEFDBL	Not required
DEFINT	Not required
DEFSNG	Not required
DEFSTR	Not available
DIM	Similar
END	Identical
ERL	Not available
ERR	Not available
ERROR	Not available
FOR/TO/STEP	Similar
GOTO	Identical
GOSUB	Identical
IF/THEN/ELSE	Similar
LET	Identical
NEXT	Similar
ON ERROR GOTO	Not available
ON/GOSUB	Identical
ON/GOTO	Similar
POINT	CALL CHAR and CALL GCHAR
POKE (Graphics)	CALL CHAR and CALL HCHAR
RANDOM	RANDOMIZE
REM	Similar
RESET	CALL CHAR and CALL HCHAR
RESUME	Not available
RETURN	Identical
SET	CALL CHAR and CALL HCHAR
STOP	STOP or BREAK

Mr. Templeton, employed in the computer industry for 20 years, has been writing computer software manuals for the past ten years.

Table 2 — I/O Statements

TRS-80 Level II	TI 99/4
DATA	Identical
INPUT	Similar
PRINT	Similar
READ	Identical
RESTORE	Identical

in four categories. For each statement or function the tables show that it is either identical, similar, not required, not available or has a substitute in TI 99/4 Basic. An identical statement or function can be used as is; 44% of the statements and functions are identical. Similar statements and functions include those that need slight changes and those that do not work exactly the same. The group of similar statements constitutes 15% of the total. There are reasonable substitutes for another 24%; that is, statements and functions that do the same things, but are written differently. Altogether, 83% of the statements and functions can be made to work on the TI 99/4. That's good transportability!

The commands are not really transported, since they are not part of the program — except those that are used as statements. However, TI 99/4 commands are at least as compatible with Level II Basic as the statements. If you are familiar with Level II commands, you will have no trouble. If you are a novice, you can learn the TI 99/4 commands as easily as any. The TI computer features some useful commands that the TRS-80 does not have.

General Requirements

The first thing you have to do to a TRS-80 program is to put each statement and function on a separate line. In the TI 99/4, the colon is used in I/O statements. No substitute is defined to separate statements on the same line. Multi-statement lines are not allowed.

The variables in your TRS-80 program will work, unless they are more than 15 characters long. Truncate any long variables to 15 characters, and any long string variables to 14 characters and $. (The strings are not limited to 15 characters, but to 112 characters.) All characters of TI 99/4 variables are significant; FIRST and FIFTH are two different variables.

An array and a variable cannot have the same name in TI 99/4 Basic. That is, if variable A is used in a program, array A is not allowed. String A$ is valid, but if string A$ is used, string array A$ may not be used.

Statements and functions listed in the tables as not required should be left out of the program. The TI 99/4 does not need them. These include DEFDBL, DEFINT, DEFSNG, CDBL, CINT and CSNG, which define and convert variable types. TI 99/4 Basic has one numeric variable type: single precision, providing 10 digits of accuracy. Type definition and conversion do not apply. The TI 99/4 does not reserve space for string variables, so CLEAR and FRE do not apply, either.

Two operators in TI 99/4 Basic use different characters from TRS-80 Level II. The string operator plus sign (+) must be replaced by an ampersand (&). The up arrow (↑) used for exponentiation must be replaced by a caret (Λ).

The TI 99/4 requires spaces that may be omitted in Level II Basic. Specifically, one of the following must immediately precede and follow each reserved word: a space; an arithmetic operator (+, −, *, /, Λ); the string operator (&); a relational operator (<, >, =), parenthesis, or other special character; the end of line character (ENTER).

Cassette Files

TI 99/4 Basic supports files on audio cassettes. You do not need any additional hardware (except the tape recorders) to use two, one for reading and one for writing. Or you can write two files concurrently.

TRS-80's INPUT # and PRINT # statements require minor changes. Remove the hyphen between the pound sign (#) and the numeral, and replace the comma after the numeral with a colon. For example, the TRS-80 statements are:

```
90 INPUT #-1,A
100 PRINT #-1,R
```

The TI 99/4 equivalents are:

```
90 INPUT #1:A
100 PRINT #1:R
```

You must add an OPEN statement before any INPUT # or PRINT # statements. The statement assigns a cassette recorder to the device number and provides messages to direct preparation of the recorder for input or output. The correct format of the OPEN statement for an input file is:

```
10 OPEN #1:"CS1",INPUT,FIXED
```

This statement says that your INPUT #1 statements apply to cassette recorder CS1; the recorder is used for input; and the records are fixed length. The OPEN statement replaces any TRS-80 PRINT statements that display cassette set-up messages.

After the file has been read, it should be closed: 500 CLOSE #1. To write to the same file, open it again: 510 OPEN #1:"CS1",OUTPUT,FIXED. This statement says that your PRINT #1 statements apply to cassette recorder CS1; the recorder is used for output; and the records are fixed length.

Only CS1 can be used for input. However, this does not mean that INPUT #-2 statements must be changed to INPUT #1. Change them to INPUT #2 and open CS1 as device 2:

```
10 OPEN#2:"CS1",INPUT,FIXED
```

TRS-80 files are not transportable to

Table 3 — String Functions

TRS-80 Level II	TI 99/4
ASC	Identical
CHR$	Identical
FRE	Not required
INKEY$	CALL KEY
LEFT$	SEG$
LEN	Identical
MID$	SEG$
RIGHT$	SEG$
STR$	Identical
STRING$	Not available
VAL	Similar

the TI 99/4. That is, files written on the TRS-80 probably will not be read correctly on the TI 99/4.

REM Statement

The TI 99/4 displays 28 characters on a line. This feature may affect a REM statement in a TRS-80 program moved to the TI 99/4. A remark longer than 28 characters continues on the next line. You may insert spaces to avoid breaking the line in the middle of a word. You are allowed 112 characters altogether in a remark statement. You may not use an apostrophe to represent the keyword REM on a TI machine. Each REM statement must occupy its own line of the program.

DEFSTR Statement

TI 99/4 does not accept the DEFSTR statement. You must use the $ explicitly in each string variable.

DIM Statement

Format of the DIM statement is the same for either computer, but arrays are limited to three dimensions in the TI 99/4. Arrays with more than three dimensions are not common. If you need them, there are at least two ways to make three-dimensioned arrays out of four-dimensioned arrays. The following TRS-80 statement defines a four-dimensioned array:

```
10 DIM A(3,4,4,4)
```

A replacement for the TI 99/4 is:

```
10 DIM A0(4,4,4), A1(4,4,4),
   A2(4,4,4), A3(4,4,4)
```

The references also have to be changed. For example, replace A(1,1,1,1) with A1(1,1,1); A(2,1,1,1) with A2(1,1,1); and A(3,1,1,1) with A3(1,1,1).

Another way is to combine two dimensions. The DIM statement for the TI 99/4 is 10 DIM A(19,4,4). The references require an expression to compute the first subscript. For example, replace TRS-80 reference A(W,X,Y,Z) with A((W+X*4),Y,Z). The multiplier for X must be 4 because 0 is always a valid subscript in the TRS-80.

ON/GOTO Statement

The difference in the ON/GOTO statement is the way in which the statement works when the value of the controlling variable or expression is equal to 0 or greater than the number of line numbers in the statement. Level II Basic falls through to the next statement in either case. The TI 99/4 gives you an error message. These two statements provide different results in the two computers:

```
150 ON X GOTO 200,300,400,500
160 GOTO 600
```

The TRS-80 executes the statement on line 160 when X is equal to 0 or greater than 4 (but not greater than 255). The TI 99/4 does not execute statement 160. Values of X other than 1, 2, 3 and 4 cause an error message and halt the program. The fix is easy:

```
143 IF X=0 THEN 600
147 IF X>4 THEN 600
150 ON X GOTO 200,300,400,500
```

FOR/TO/STEP Statement

The FOR/TO/STEP statement of TI 99/4 Basic works the same as that of Level II Basic in most cases. However, the TRS-80 always executes the loop one time, even when the initial value of the counter variable is beyond the limit. The TI 99/4 compares the initial value to the limit before performing the loop and does not perform the statements in the loop when the value is already at or beyond the limit.

Often a statement is included in the program to prevent the initial execution of the loop:

```
100 IF X>=5 THEN 150
110 FOR I=X TO 5
120 PRINT I;
130 NEXT I
150 X=X/2
```

The statement on line 100 is not needed in the TI 99/4 because the statement on line 110 branches to line 150 when X is greater than or equal to 5. If you want to print the value of X regardless of what it is, you need two more statements in TI 99/4 Basic:

```
120 IF X>=THEN 125 ELSE 130
125 PRINT X;
130 FOR I=X TO 5
135 PRINT I;
140 NEXT I
```

NEXT Statement

The NEXT statement for the TI 99/4 must specify the counter variable. The TRS-80 accepts these statements:

```
120 FOR X=1 TO 10
130 A=A+1
140 NEXT
```

To run in the TI 99/4, change line 140:

```
140 NEXT X.
```

IF/THEN/ELSE Statement

The IF/THEN/ELSE statement in TI 99/4 Basic is significantly different from that of TRS-80 Level II. Keywords THEN and ELSE may only be followed by line numbers (branching to other statements). You could have the following statements in a TRS-80 program:

```
150 IF X= 1 THEN Y=2 ELSE Y=0
160 PRINT Y;
170 END
```

For the TI 99/4, you need these statements:

```
150 IF X=1 THEN 250
155 Y=0
160 PRINT Y;
170 END
250 Y=2
260 GOTO 160
```

The multi-statement line capability of the TRS-80 allows the following:

```
150 IF X=0 THEN Y=0:Z=0:STOP
160 Y=2
```

When X is equal to zero, Y and Z are set

Table 4 — Artithmetic Functions

TRS-80 Level II	TI 99/4
ABS	Identical
ATN	Identical
CDBL	Not required
CINT	Not required
COS	Identical
CSNG	Not required
EXP	Identical
FIX	INT (Positive values)
INT	Identical
LOG	Identical
RND (n)	Available indirectly
SGN	Identical
SIN	Identical
SQR	Identical
TAN	Identical
POS	Not available
USR	Not available
VARPTR	Not available

to zero, and execution stops. When X is not equal to zero, the statements on lines 160 and 170 are executed. To do this on the TI 99/4, you need the following:

```
150 IF X=0 THEN 200
160 Y=2
170 END
200 Y=0
210 Z=0
220 STOP
```

TI 99/4 Basic has no logical operators, but you can perform logical operations. A relational operator may be used in any expression. The value of a relational expression is −1 when the relation is true, and 0 when the relation is false, thus providing logical operations with an arithmetic operator.

An example is a TRS-80 statement, as follows:

```
200 IF A$>B$ AND A$>C$ THEN
    PRINT A$
210 END
```

Replacing the AND operator with a plus sign changes the logical function to an addition operation. The only possible sums are 0, −1 and −2. The sum of −2 corresponds to the true result of an AND function. So the equivalent statements for the TI 99/4 are:

```
200 IF (A$>B$)+(A$>C$)=-2
    THEN 205 ELSE 210
205 PRINT A$
210 END
```

The OR operator can also be replaced by a plus sign; when the result is less than zero, the OR function is true. A valid example for the TRS-80 is:

```
200 IF A=B OR C=D
    THEN 250 ELSE 260
```

The TI 99/4 equivalent is:

```
200 IF (A=B)+(C=D)<0 THEN 250
    ELSE 260
```

RANDOM Statement

The TI 99/4 equivalent of the RANDOM statement is the RANDOMIZE statement. It may be used with an argument to provide a sequence of random numbers that depends upon the value of the argument. The sequence is always the same for each value of the argument.

RND(n) Function

The TI 99/4 RND function is equivalent to the RND(0) function of Level II Basic. That is, RND returns a fraction between 0 and 1. But you can define a function that is equivalent to the RND(n) function of the TRS-80.

The TI 99/4 DEF statement defines a function that can be defined in a single statement. It must be placed in the program ahead of any reference to the function, and may be referenced as often as required. The DEF statement for the random integer function is:

```
10 DEF RAND(N)=INT(N*RND)+1.
```

You cannot call the new function RND because there already is an RND function. You may use any name that is not a reserved word (or already used). An example of the use of RND(n) in the TRS-80 is: 220 CARD=RND(52). This statement gives variable CARD a different value in the range of 1 through 52 each time it is executed. With function RAND defined, the following statement does the same thing in the TI 99/4: 220 CARD=RAND(52).

FIX Function

The FIX function of TRS-80 Level II Basic returns the same result as the INT function when the argument is positive. The INT function of TI 99/4 Basic can be used as a substitute if the argument is always positive. When the argument may be either negative or positive, statements can be added to correct the result for negative values. The TRS-80 statement is:

```
350 XI=FIX(X)
```

The TI 99/4 equivalent is:

```
345 IF X<0 THEN 347 ELSE 370
347 XI=X+1
350 XI=INT(XI)
360 END
370 XI=X
380 GOTO 350
```

STOP Statement

TI 99/4's STOP statement is identical to the END statement. If you want to continue after a STOP, use the BREAK statement instead. Enter the CONTINUE command to resume execution.

CLS Statement

CALL CLEAR in TI 99/4 Basic is equivalent to the CLS statement of Level II Basic.

INPUT Statement

Replace the semicolon after the prompting message of an INPUT statement with a colon. For example, the following is valid in the TRS-80:

```
220 INPUT "PRICE";P
```

For the TI 99/4, the statement is:

```
220 INPUT "PRICE? ":P
```

The TRS-80 displays a question mark after PRICE automatically; if you want it on the TI 99/4, you have to put it in the prompting message.

INKEY$ Function

The KEY subprogram of TI 99/4 Basic replaces the INKEY$ function of TRS-80 Level II Basic. An example of the use of INKEY$ is:

```
150 ANS$=INKEY$:IF ANS$=""
    THEN 150
```

TI 99/4's equivalent is:

```
150 CALL KEY (0, ANS, ST)
160 IF ST=0 THEN 150
170 ANS$=CHR$(ANS)
```

The KEY subprogram returns zero in variable ST until a key is pressed. The subprogram places the numeric value of the character corresponding to the key in variable ANS. Statement 170 uses the CHR$ function (identical in both computers) to obtain the character.

PRINT Statement

PRINT statement differences are only a part of the problem of transporting PRINT statements from the TRS-80 to the TI 99/4. Many displays have to be redesigned because the TI 99/4 screen has 24 lines of 28 characters each, rather than TRS-80's 16 lines of 64 characters. You may want to rewrite these parts of the program anyway to include color. So you will need to know the differences.

The PRINT statement in TI 99/4 Basic may not be abbreviated. The following is valid in TRS-80:

```
130 ? A$
```

Change it, as follows, for the TI 99/4:

```
130 PRINT A$
```

The TAB modifier must be followed by a print separator in TI 99/4 Basic. The following statement is valid for the TRS-80:

```
130 PRINT TAB(5) SUM; TAB(20) DIFF
```

But for the TI 99/4, it requires print separators:

```
130 PRINT TAB(5);SUM;TAB(20);DIFF
```

Any separator works, but a comma gives strange results. The comma causes the next item printed to begin in the next print zone, as in TRS-80 Level II Basic. There are two print zones per line; one beginning at column 1, and the other beginning at column 15. So when the TAB modifier is followed by a comma, the item is displayed beginning either at column 15 or at column 1 of the next line. The semicolon should follow a TAB modifier as in the example.

TI 99/4 Basic also uses the colon as a separator. It causes the next item to be displayed on the next line. You can use two of them to skip a line. A TRS-80 program might contain this statement:

```
200 PRINT "A","B":PRINT:PRINT A,B
```

You can do the same thing with one PRINT statement in the TI 99/4:

```
200 PRINT "A","B"::A,B
```

The PRINT @ statement of TRS-80 Level II Basic is not available in TI 99/4 Basic, but there is an HCHAR sub-

program that prints a character at any location on the screen. The subprogram requires a row number, a column number and the decimal value of the character to be displayed. You can use the HCHAR subprogram, one SEG$ function, the LEN function and the ASC function to duplicate the PRINT @ statement. A TRS-80 example is:

```
200 A$="YOUR NAME"
210 CLS
220 PRINT @128,A$
```

The TI 99/4 equivalent is:

```
200 A$="YOUR NAME"
205 CALL CLEAR
210 FOR I=1 TO LEN(A$)
215 X=ASC(SEG$(A$,I,1))
220 CALL HCHAR(3,2+I,X)
225 NEXT I
```

The LEN and ASC functions are the same in both computers. The SEG$ function is similar to the MID$ function of Level II Basic. Statement 215 assigns the decimal value of one character to variable X. Statement 220 displays the character in a specified position. TRS-80 position 128 is the first character position on line 3. The first argument for subprogram HCHAR is 3, which specifies line 3 on the TI 99/4 screen. The second argument, 2+I, places the first character in the first character position on the line. The FOR/TO/NEXT loop is executed for each character of string A$.

The PRINT USING statement for formatted output is also not available in TI 99/4 Basic. However you can do some of the same things in different ways. The pound sign (#) field specifier is used to round off a number to dollars and cents, as in the following example:

```
230 PRINT USING "###.##"; PAY
```

If PAY is a value such as 238.1368, it is displayed 238.14. You can use a DEF statement in TI 99/4 Basic to define a rounding function and reference that function in PRINT statements as you need it. The DEF statement is:

```
10 DEF RND2(X)=INT(X*100+.5)/100
```

To round to three decimal places substitute 1000 for 100. You can substitute 10 for 100 and round to one decimal place. Reference the function in PRINT statements as follows:

```
230 PRINT RND2(PAY)
```

VAL Function

The VAL function is similar in both computers. However, the TRS-80 function evaluates the numeric portion of a mixed string consisting of a number followed by alphanumeric characters. The VAL function in TI 99/4 Basic evaluates a number only. The following statements are valid for the TRS-80:

```
400 A$="429 WIDGETS"
410 A=VAL(A$)
```

Statement 410 assigns 429 as the value of A; WIDGETS is not evaluated. To do the same thing in the TI 99/4, use the POS and SEG$ functions. The POS function locates the space following the number, the SEG$ function forms a substring that contains the number. The substitute statements are:

```
400 A$="429 WIDGETS"
405 B$=SEG$(A$,1,POS(A$," ",1)-1)
410 A=VAL(B$)
```

LEFT$, MID$, RIGHT$ Functions

The LEFT$, MID$, and RIGHT$ functions of TRS-80 Level II Basic can be replaced by the SEG$ function of TI 99/4 Basic. SEG$ requires the same three arguments as MID$. LEFT$ and RIGHT$ each require only two arguments, one short of the three required for SEG$.

An example of the use of the LEFT$ function is:

```
400 PRINT LEFT$(A$,5)
```

When replacing the LEFT$ function with the SEG$ function in TI 99/4 Basic, the second argument is always 1:

```
400 PRINT SEG$(A$,1,5)
```

An example of the use of MID$ is:

```
420 PRINT MID$(A$,5,3)
```

The TI 99/4 equivalent function uses the same arguments:

```
420 PRINT SEG$(A$,5,3)
```

An example of the use of the RIGHT$ function is:

```
440 PRINT RIGHT$ (A$,4)
```

The second argument of the SEG$ function to replace the RIGHT$ function must be computed. The statements for the TI 99/4 are:

```
440 M=LEN(A$)-3
445 PRINT SEG$(A$,M,4)
```

Statement 440 subtracts 3 from the length of the string to compute the position of the fourth character from the end of the string. If the length of the desired substring were variable N, statement 440 would subtract N-1 from the length.

STRING$ Function

The STRING$ function of the TRS-80 can be replaced by a FOR/TO/NEXT loop in the TI 99/4. The following TRS-80 statement assigns a string of hyphens to string variable M$:

```
300 M$=STRING$(15,"-")
```

The following statements assign a string of 15 hyphens to string variable M$ in the TI 99/4:

```
300 M$=""
310 FOR N=1 TO 15
320 M$=M$&"-"
330 NEXT N
```

These statements are the equivalent of the STRING$ function. However, when the STRING$ function is used in a PRINT@ statement, a call to the HCHAR subprogram using the repetition argument would replace the entire statement. The TRS-80 statement is as follows:

```
450 PRINT @128 STRING$(15,"-")
```

The TI 99/4 statement is:

```
450 CALL HCHAR (3,3,45,15)
```

The first two arguments are both three, specifying the first character position on the third line. This corresponds to position 128 on the TRS-80 screen. The third argument is the numeric character code, 45 for a hyphen. The fourth argument is the number of repetitions, 15 in the example.

Graphics

Comparison of the statements, functions and subprograms for graphics becomes difficult because of the differences in screen layout. The SET and RESET statements control 6144 discrete blocks on the screen of the TRS-80. The CHAR and HCHAR subprograms control 49,152 discrete blocks on the screen of the TI 99/4. TRS-80's SET and RESET statements address the blocks on the screen with horizontal and vertical coordinates. TI's HCHAR subprogram addresses each of 768 character positions by row and column. The 64 discrete blocks within a character position are set or reset by the CHAR subprogram. You have eight times the resolution of the TRS-80 Level II graphics available on the TI 99/4.

The horizontal dimension of the screen is divided into 128 blocks, two per character position, on the TRS-80. The same dimension is divided into 256 blocks on the TI 99/4, eight per character position. The TRS-80 divides the vertical dimension of the screen into 48 blocks, three per line. The TI 99/4 divides the same dimension into 192 blocks, eight per line. Thus a TI 99/4 character position contains eight TRS-80 blocks, with the top half divided into four blocks, and the bottom half divided into four more blocks. Each block is four TI 99/4 blocks high and two TI 99/4 blocks wide.

The CHAR subprogram defines a character with any possible combination of set and reset states of the 64 blocks in the character position. The following statements define eight characters, each setting one TRS-80 block

within the character position and resetting the other seven:

```
10 DATA "COCOCOCO", "30303030"
20 DATA "0COCOCOC","03030303"
30 DATA "00000000COCOCOCO"
40 DATA "0000000030303030"
50 DATA "000000000COCOCOC"
60 DATA "0000000003030303"
70 FOR I=1 TO 8
80 READ C$
90 CALL CHAR(100+I,C$)
100 NEXT I
```

The READ statement reads a string and the CHAR subprogram defines character 101 according to the contents of the string. The pattern that corresponds to character 101 sets a TRS-80 block in the upper left corner of a character position. The loop causes the statements to be performed eight times. Each of the patterns sets one TRS-80 block.

The TRS-80 X and Y coordinates can be converted to a row, column and character for the HCHAR subprogram. The following TRS-80 statement sets one block on the screen: 140 SET(X,Y).

Using the characters defined in the preceding example, the following TI 99/4 statements set the same block:

```
110 DEF RINC(Y)=(Y-INT(Y/2)*2)
120 DEF CINC(X)=(X-INT(X/4)*4)+1
140 R=INT(Y/2)+1
150 C=INT(X/4)+1
160 N=CINC(X)+(RINC(Y)*4)+100
170 CALL HCHAR(R,C,N)
```

The DEF statements define functions RINC and CINC for use in computing N. Function RINC computes a row increment, which is the remainder when the argument is divided by 2. The column increment, CINC, is one greater than the remainder when the argument is divided by 4. Statement 140 assigns to R the value of the integer quotient of Y/2 plus 1. The increment of 1 is necessary because the TI 99/4 numbers rows and columns starting at 1. Statement 150 assigns to C the value of the integer quotient of X/4 plus 1. Statement 160 computes a number that corresponds to the value of one of the defined characters. The number is the sum of the column increment (left over when converting the X coordinate to a column), four times the row increment (left over when converting the Y coordinate to a row) and 100, the base number of the character set. The characters set the TRS-80 block that corresponds to these increments. Statement 170 calls the HCHAR subprogram to display the character, setting the indicated block on the screen.

To reset the block, execute the following statement:

```
200 CALL HCHAR(R,C,32).
```

Similar characters can be defined to correspond to any combination of SET and RESET statements. However, in many cases it is better to redesign the display to exploit the capabilities of the TI 99/4.

There is no POINT function in TI 99/4 Basic, but the GCHAR subprogram reads the character in a specified character position. You could replace line 170 in the preceding example with the following: 170 CALL GCHAR (R,C,CH). If none of the eight TRS-80 blocks had been set in character C on row R, variable CH would have a value of 32 (space). If one of the blocks had been set, CH would have a value corresponding to the character value.

You *can* transport a TRS-80 Level II Basic program to the TI 99/4 computer. Some things have to be changed a bit. Displays may have to be redesigned because of the difference in screen layouts. TI 99/4 equivalents of TRS-80 programs tend to require more lines because the TI 99/4 allows only one statement per line. But much of the TRS-80 program will run on the TI 99/4 with no change. □

Free-Form Storage and Retrieval System

BY WILLIAM LAPPEN

Do you have separate programs to keep track of your record collection, computer programs, recipes, mailing lists and so forth? While these problems all require special title headings and special retrieval procedures, it is possible to do them all in a single general purpose program.

The system explained in this article was inspired by IBM's System 6 computer (over $15,000) and is implemented on a 48K TRS-80 with disk (less than $2000). (If you don't have a disk, just change the #1 to #-1 in the input and output statements. You will also need to delete the file open and close commands and change LINE INPUT to INPUT.) I certainly don't claim that the program is as versatile as System 6, but then again, it ain't that far from it!

The program allows free-form entry of information into categories that you specify and name. After entry, you may retrieve the information in a number of different ways. For example, let's assume that we want to catalog all of our recipes in the data base. We decide that there are three major characteristics of importance to us for each recipe (Type of Dish, Major Ingredient, Complexity).

After entering all of our menus into the data base, we can retrieve them in some very useful ways. Let's assume we want to invite some people over for dinner and let's plan the menu. First we need an entree. We search the TYPE OF DISH category for ENTREE. Having found 15 entrees, we select "Hamburger". Now we need a side dish. For a side dish, we want to serve a noodle dish. We search the data base only for MAJOR INGREDIENTs that are NOODLE and search TYPE OF DISH for dishes that are SIDE dishes. This search produces five possibilities and we select one. Now for dessert, we really want to astound our friends. We search COMPLEXITY for VERY and TYPE OF DISH for DESSERT.

Notice that we can search by any of the categories we have entered into the data base. In addition, we may search for a combination of attributes across categories (side dish with noodles).

The same data base manipulation program will allow us to catalog all of our records and retrieve by title, group, type of music and so forth. If we wanted to retrieve everything that was not rock music, we could even do that. Further, we may have the resulting list alphabetized by any category.

What about using it as an address book? Obviously, you can alphabetize by last name. But have you ever considered sorting your address list by city or state? Perhaps a business mailing list would benefit by such a retrieval (a special sale in only one of the business' locations — no need to tell the whole world, only local customers are interested). The applications of such a free-form data base are practically unlimited.

Overall Structure

The program takes full advantage of literal (or string) arrays. These let you enter as much or as little information as desired with the only limit being 255 characters per sub-entry. Sequential access files are also used for the same reason. Random access files require you to define field sizes. Since the whole point of the program is to eliminate the need to constrain the fields, this would not be satisfactory. Since all manipulations are handled in RAM, the faster record retrieval of random access files is of no particular use.

The menu in Figure 1 is your master control for the program. Let's work

```
READY                                    COLUMN TITLES
>RUN
STORAGE AND RETRIEVAL SYSTEM              1 TITLE
                                          2 F. NAME
                                          3 L. NAME
1  READ INFORMATION FROM DISK             4 ADDRESS
                                          5 CITY
2  TYPE IN INFORMATION                    6 STATE
                                          7 ZIP
3  SORT INFORMATION                       8 AMT DUE

4  RECORD INFORMATION ON DISK

5  CHANGE INFORMATION

YOUR CHOICE ? 2
NUMBER OF COLUMNS ? 8
```

Figure 1

```
          ADD INFORMATION
            ID NUMBER  1

1  TITLE            MR.
2  F. NAME          JOHN
3  L. NAME          JONES
4  ADDRESS          101 ANYWHERE STREET
5  CITY             ANYTOWN
6  STATE            CA
7  ZIP              9569
8  AMT DUE          100.54

          CORRECT (Y/N/X) ? Y
```

Figure 2

through the program entering an address list with the amount of money each person owes. First, we need to type in the information (number 2 on the list). If we had previously entered and saved the information, we could merely read it from the disk (number 1) and change information (number 5) or add information (number 2).

Since there is no information in the computer, we must decide on category titles. We figure that there are eight pieces of information that are relevant for each person. They are entered in Figure 2. Now we are ready to enter the address list. The computer will tell us the record ID NUMBER and prompt with the category titles we have entered. After entering the information, the computer asks whether it is CORRECT (Y/N/X). If correct, type Y and the computer is ready for the next record. If you made a mistake on one of the lines, type its number and you can change that entry. If the information is hopeless, type N and the computer will let you reenter the whole thing.

When you are done with the entry phase, type END for the first category (TITLE in our case.) This will return you to the menu. From the menu, let's look at the SORT INFORMATION routine (number 3). I have entered three different people into the data base so we will have something to play with.

The computer will ask for a SEARCH REQUEST. At this point, you specify the category, relation and object of the search. The category may be specified by number or actual title. For example, the first search in Figure 3 is L. NAME = ALL. This instructs the computer to search the L. NAME category (number 3). In that category, it is looking for "=" (as opposed to "<", ">", or "< >"). In this case the object, ALL, is a magic word meaning just print all entries.

Having found all entries that fit the search request, the computer is ready to sort them. We tell it to sort by column 2 (F. NAME). Then it is ready to print the results for us. We don't want to print all eight categories that we entered, only a few. All that we want on this search is the TITLE, F. NAME, L. NAME and AMT DUE. Again, these categories may be entered by name or number. The computer complies. Notice that the ID number is printed for each record. This is an internal number that becomes slightly useful for making changes. Also notice that the list is alphabetized by F. NAME as requested.

Let's try another retrieval. Assume we want to retrieve all entries that are not doctors. We type TITLE < > DR. and the computer will do the retrieval. (If we wanted all non-doctors who lived in California, we would type TITLE < > DR. AND STATE = CA. You may string together as many restrictions as you want in one search request.) Now the list is sorted by L. NAME and printed. When you are done with the sort, type END for the SEARCH REQUEST.

To make changes in the records,

```
SEARCH REQUEST ? L. NAME = ALL
SORT BY 2
COLUMNS TO PRINT 1,2,3,AMT DUE
 ID  TITLE          F. NAME        L. NAME        AMT DUE

 3   MS.            GLORIA         NELSON         10.45
 1   MR.            JOHN           JONES          100.54
 2   DR.            SAM            SAMPSON        56.66


 SEARCH REQUEST ? TITLE <> DR.
SORT BY L. NAME
COLUMNS TO PRINT F. NAME,3,8
 ID  F. NAME              L. NAME              AMT DUE

 1   JOHN                 JONES                100.54
 3   GLORIA               NELSON               10.45
```

Figure 3

```
                                    10427 Lindbrook Drive
                                    Los Angeles, CA  90024
                                    May 9, 1980

Mr. John Jones
101 Anywhere street
Anytown, DE  86345

Dear Mr. Jones,

     As you know, John, we are in the midst of a
severe credit crunch.  Our suppliers are reluctant to
extend credit to us and we must therefore request
on your account of $100.54.  We sincerely hope that
this is merely a temporary situation and look forward
to serving you in the future.

     Thank you for your cooperation.

                    Sincerely,

-------------------------------------------------------

                                    10427 Lindbrook Drive
                                    Los Angeles, CA  90024
                                    May 9, 1980

Dr. Sam Sampson
431 54th Street
Beverly woods, CO  56214

Dear Dr. Sampson,

     As you know, Sam, we are in the midst of a
severe credit crunch.  Our suppliers are reluctant to
extend credit to us and we must therefore request
on your account of $56.66.  We sincerely hope that
this is merely a temporary situation and look forward
to serving you in the future.

     Thank you for your cooperation.

                    Sincerely,
```

Figure 4

merely enter number 5 from the menu. You may retrieve a record by ID NUMBER or any of the categories. For example, if you wanted to retrieve by TITLE, merely hit a carriage return to null through the ID NUMBER. The cursor moves down to the TITLE category. Enter your request and the computer shows you the first record that fulfills that request. If you want to delete the entry, type D. If you want to change a category entry, type its number. If you like the record the way it is, type Y. If you didn't even want to see that record, just hit the carriage return and the computer will give you the next record that fulfills your search request. This makes it very easy to retrieve a record to change it. You only need to know one piece of information about the record (the ID NUMBER or the contents of any category). When done with the change routine, type END as the attribute to find. You'll find yourself gazing at the menu again.

To save the information you have so diligently typed in, type a number 4. If you have retrieved some information from the disk, the new information will replace it. If this is new information, you'll have to come up with some cute file name so the computer can store the information for you.

Interfacing With Other Programs

While you have probably seen something of value in the data base storage and retrieval system presented here, the real power comes when other programs are tied to it. The output from the program is a sequential file that is very easy to read and manipulate. (See program lines 860 to 1000 for the code to read the file.)

As an example of using the output from the storage program as input for other programs, let's continue with that list of people we entered and saved. Now, let's send them personalized letters reminding them that they owe us money. The program allowing the entry of text is titled AUTO-WRITE, a very simple wordprocessor intended to illustrate the concept of merging text and data.

First, enter and clean up the text for the letter. Wherever you want to add a personal touch, just type the category title. For example, typing "Dear TITLE L. NAME," will be typed out "Dear Mr. Jones," in the first letter. The computer searches through the text for the title and replaces it with the individual entry. (See lines 610 to 690.) After the letter is printed, the process is reversed (lines 750 to 830). Another approach would be to search the letter for the I-1 entry and replace it with the I entry (starting at I=1). Figure 4 shows the first two letters printed from the data base previously entered.

Other Uses

What about putting real estate listings in the data base? When a client wants to see what houses are available in a certain area with two or more bedrooms and two or more bathrooms, that information can be retrieved. Or how about using it to inventory parts on hand in a manufacturing operation? The only real limit to the applications of the free-form storage and retrieval system is your own imagination. □

Mr. Lappen spends much of his free time creating progams to help solve business and management problems. His previous PC *articles include "Check Entry and Retrieval System" (June 1980) and "Stock Option Analysis" (August 1980).*

Auto-Write Program Listing

```
10 'AUTO-WRITE FOR SORT 5/9/80
20 CLEAR 22000
30 DIM A$(100)
40 S$=CHR$(23)
50 J=0
60 CLS
70 PRINT S$
80 PRINT TAB(10) "AUTO-WRITE"
90 PRINT
100 PRINT "1 READ LIST FROM DISK"
110 PRINT "2 READ TEXT FROM DISK"
120 PRINT "3 TYPE IN TEXT"
130 PRINT "4 PRINT/MERGE TEXT AND LIST"
140 PRINT "5 SAVE TEXT ON DISK"
150 PRINT
160 PRINT
170 PRINT TAB(10) "YOUR CHOICE ";
180 INPUT A
190 IF A<1 OR A>5 GOTO 60
200 ON A GOTO 210  ,340  ,420  ,590  ,870
210 'READ LIST FROM DISK
220 INPUT "FILE NAME";F$
230 OPEN "I",1,F$
240 INPUT#1,K
250 R=1000/K
260 DIM L$(R,K)
270 R=0
280 FOR I=1 TO K
290 LINE INPUT#1,L$(R,I)
300 NEXT I
310 IF EOF(1) CLOSE: GOTO 60
320 R=R+1
330 GOTO 280
340 'READ TEXT FROM DISK
350 INPUT "TEXT FILE NAME ";N$
360 OPEN "I",1,N$
370 J=1
380 LINEINPUT #1, A$(J)
390 IF EOF(1) CLOSE: GOTO 60
400 J=J+1
410 GOTO 380
420 'TYPE IN TEXT FROM KEYBOARD
430 CLS
440 J=J+1
450 LINEINPUT A$(J)
460 IF A$(J)="END" THEN J=J-1: GOTO 480
470 GOTO 440
480 CLS
490 FOR I=1 TO J
500 PRINT I; A$(I)
510 NEXT I
520 INPUT "CORRECT (Y/#) ";A$
530 IF A$="Y" GOTO 60
540 A=VAL(A$)
550 IF A<1 OR A>J GOTO 520
560 PRINT A$(A)
570 LINEINPUT A$(A)
580 GOTO 480
590 'PRINT/MERGE TEXT AND LIST
600 K1=1
610 'REPLACE CATEGORIES WITH SPECIFICS
620 FOR I=1 TO K
630 FOR II=1 TO J
640 S=INSTR(A$(II),L$(0,I))
650 IF S=0 GOTO 680
660 A$(II)=LEFT$(A$(II),S-1)+L$(K1,I)+
    RIGHT$(A$(II),LEN(A$(II))-LEN
    (L$(0,I))-S+1)
670 GOTO 640
680 NEXT II
690 NEXT I
700 'READY TO PRINT
```

```
710 INPUT "PAPER IN PRINTER ";A$
720 FOR I=1 TO J
730 LPRINT A$(I)
740 NEXT I
750 'REPLACE SPECIFICS WITH CATEGORIES
760 FOR I=1 TO K
770 FOR II=1 TO J
780 S=INSTR(A$(II),L$(K1,I))
790 IF S=0 GOTO 820
800 A$(II)=LEFT$(A$(II),S-1)+L$(0,I)+
    RIGHT$(A$(II),LEN(A$(II))-LEN(L$
    (K1,I))-S+1)
810 GOTO 780
820 NEXT II
830 NEXT I
840 K1=K1+1
850 IF K1<=R GOTO 620
860 GOTO 60
870 'SAVE TEXT ON DISK
880 INPUT "TEXT FILE NAME ";N$
890 OPEN "O",1,N$
900 FOR I=1 TO J
910 PRINT #1, A$(I)
920 NEXT I
930 CLOSE
940 GOTO 60
```

Sort Program Listing

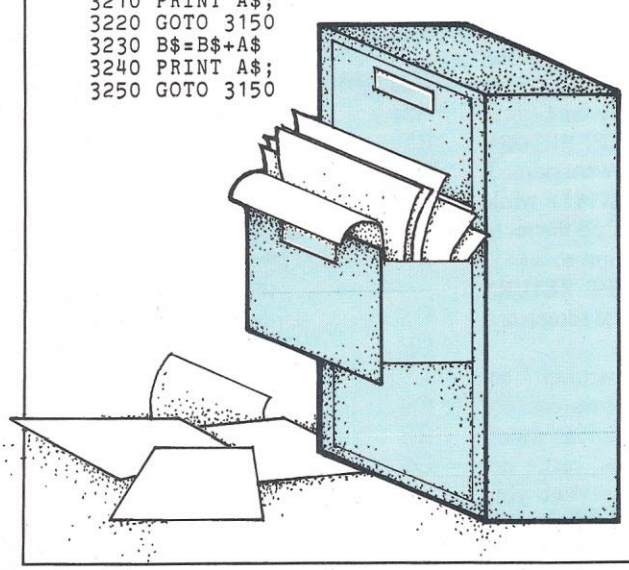
```
10 'SORT 5/9/80
20 GOTO 100
30 PRINT "OUT OF MEMORY"
40 FORII=1TOK
50 L$(R,II)=""
60 NEXT
70 R=R-1
80 INPUT "ENTER WHEN READY";A$
90 RESUME 140
100 CLEAR 22000
110 DEFINT A-Z
120 E$=CHR$(30)
130 S$=CHR$(23)
140 CLS
150 PRINT S$;
160 F2=0
170 ON ERROR GOTO 30
180 PRINT "STORAGE AND RETRIEVAL SYSTEM"
190 PRINT
200 PRINT
210 PRINT "1  READ INFORMATION FROM DISK"
220 PRINT
230 PRINT "2  TYPE IN INFORMATION"
240 PRINT
250 PRINT "3  SORT INFORMATION"
260 PRINT
270 PRINT "4  RECORD INFORMATION ON DISK"
280 PRINT
290 PRINT "5  CHANGE INFORMATION"
300 PRINT
310 PRINT
320 INPUT "YOUR CHOICE ";I
330 IF I<1 OR I>5 GOTO 320
340 ON I GOTO 860  , 350  , 1140 , 1010 ,
    2520
350 'TYPE IN INFORMATION
360 IF R>0 GOTO 490
370 INPUT "NUMBER OF COLUMNS ";K
380 IF K<0 OR K>11 GOTO 370
390 I=1000/K
400 DIM L$(I,K),N(I),S$(K),R(K),C(K)
410 CLS
420 PRINT S$;
430 PRINT "COLUMN TITLES"
440 PRINT
450 FOR I=1 TO K
460 PRINT I;
470 LINE INPUT L$(0,I)
480 NEXT
490 JS=60/K
500 KK=0
510 FOR I=1 TO K
520 IF LEN(L$(0,I))>KK THEN KK=LEN(L$(0,I))
530 NEXT
540 KK=KK*2+12
550 IF F2<0 GOTO 140
560 CLS
570 PRINT S$;
580 PRINT TAB(6) "ADD INFORMATION"
590 R=R+1
600 PRINT TAB(8) "ID NUMBER "; R
610 PRINT
620 FOR I=1 TO K
630 PRINT I; " "; L$(0,I)
640 NEXT
650 F=0
660 I=1
670 J1=192
680 GOSUB 3110
690 L$(R,I)=B$
700 IF L$(R,I)="END" GOTO 830
710 IF F=1 GOTO 740
720 I=I+1
730 IF I<=K GOTO 670
740 'VERIFY ENTRY
750 F=1
760 PRINT @ 908, "CORRECT (Y/N/X) ";
770 INPUT A$
780 IF A$="Y" GOTO 560
790 IF A$="N" THEN R=R-1: GOTO 560
800 I=VAL(A$)
810 IF I<=0 OR I>K GOTO 760
820 GOTO 670
830 'DONE WITH ENTRY
840 R=R-1
850 GOTO 140
860 'READ INFORMATION FROM DISK
870 IF R>0 PRINT "ALREADY HAVE INFORMATION
    IN BUFFER":GOTO 320
880 INPUT "FILE NAME";F$
890 OPEN "I",1,F$
900 INPUT#1,K
910 R=1000/K
920 DIM L$(R,K),N(R),S$(K),R(K),C(K)
930 JS=60/K
940 R=0
950 FOR I=1 TO K
960 LINE INPUT#1,L$(R,I)
970 NEXT
980 IF EOF(1) CLOSE: F2=-1: GOTO 490
990 R=R+1
1000 GOTO 950
1010 'WRITE INFORMATION TO DISK
1020 IF R=0 PRINT "NOTHING IN BUFFER": GOTO 320
1030 IF F$<>"" GOTO 1050
1040 INPUT "FILE NAME";F$
1050 OPEN"O",1,F$
1060 PRINT#1,K
1070 FOR I=0 TO R
1080 FOR II=1 TO K
1090 PRINT#1,L$(I,II)
1110 NEXT
1120 CLOSE
1130 GOTO 140
```

continued

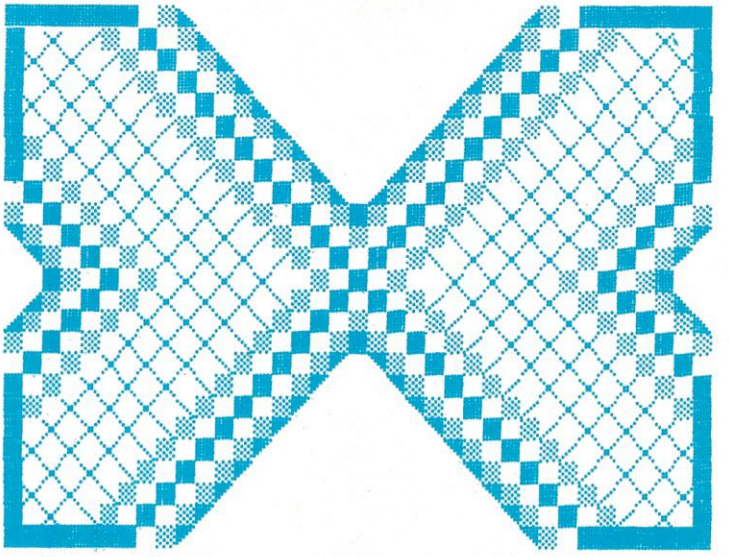
Sort Program Listing continued

```
1140 'SELECT AND SORT INFORMATION
1150 FOR I=1 TO R
1160 N(I)=0
1170 NEXT
1180 'INPUT CHARACTERISTICS
1190 CLS
1200 INPUT "SEARCH REQUEST ";A$
1210 IF A$="END" GOTO 140
1220 SI=1
1230 II=1
1240 J=ASC(MID$(A$,II,1))
1250 IF J>59 AND J<63 GOTO 1310
1260 II=II+1
1270 IF II<LEN(A$) GOTO 1240
1280 'ERROR
1290 PRINT "INVALID REQUEST"
1300 GOTO 1200
1310 B$=LEFT$(A$,II-1)
1320 IF LEFT$(B$,1)=" " B$=RIGHT$
     (B$,LEN(B$)-1): GOTO 1320
1330 IF RIGHT$(B$,1)=" " B$=LEFT$
     (B$,LEN(B$)-1):GOTO 1330
1340 'GET COLUMN
1350 FOR J1=1 TO K
1360 IF B$=L$(0,J1) GOTO 1410
1370 NEXT J1
1380 C(SI)=VAL(B$)
1390 IF C(SI)>0 AND C(SI)<=K GOTO 1420
1400 PRINT B$; " DOES NOT EXIST": GOTO 1200
1410 C(SI)=J1
1420 'GET RELATIONSHIP
1430 R(SI)=0
1440 R(SI)=R(SI)+10[(J-60)
1450 II=II+1
1460 IF II>LEN(A$) GOTO 1280
1470 J=ASC(MID$(A$,II,1))
1480 IF J>59 AND J<63 GOTO 1440
1490 IF J=32 GOTO 1450
1500 A$=RIGHT$(A$,LEN(A$)-II+1)
1510 II=1
1520 J=0
1530 IF MID$(A$,II,1)<>"&" GOTO 1570
1540 S$(SI)=LEFT$(A$,II-1)
1550 II=II+1+J
1560 GOTO 1610
1570 IF MID$(A$,II,3)="AND" THEN J=2:
     GOTO 1540
1580 II=II+1
1590 IF II<LEN(A$) GOTO 1520
1600 S$(SI)=LEFT$(A$,II)
1610 IF RIGHT$(S$(SI),1)=" "
     S$(SI)=LEFT$(S$(SI),LEN(S$(SI))-1):
     GOTO 1610
1620 IF II>=LEN(A$) GOTO 1660
1630 A$=RIGHT$(A$,LEN(A$)-II)
1640 SI=SI+1
1650 GOTO 1230
1660 'SEARCH
1670 I8=1
1680 II=0
1690 FOR J=1 TO SI
1700 IF S$(J)="ALL" GOTO 1810
1710 I7=R(J)
1720 IF I7<100 GOTO 1750
1730 IF L$(I8,C(J))>S$(J) GOTO 1810
1740 I7=I7-100
1750 IF I7<10 GOTO 1780
1760 IF L$(I8,C(J))=S$(J) GOTO 1810
1770 I7=I7-10
1780 IF I7<1 GOTO 1800
1790 IF L$(I8,C(J))<S$(J) GOTO 1810
1800 GOTO 1850
1810 NEXT J
1820 II=II+1
1830 N(II)=I8
1840 GOTO 1800
1850 I8=I8+1
1860 IF I8<=R GOTO 1690
1870 'DONE
1880 'READY TO SORT AND PRINT
1890 LINEINPUT "SORT BY ";A$
1900 FOR J=1 TO K
1910 IF A$=L$(0,J) GOTO 1950
1920 NEXT J
1930 J=VAL(A$)
1940 IF J=0 GOTO 1890
1950 C(0)=J
1960 LINEINPUT "COLUMNS TO PRINT ";A$
1970 SI=1
1980 J=INSTR(A$,",")
1990 IF J=0 GOTO 2060
2000 S$(SI)=LEFT$(A$,J-1)
2010 IF RIGHT$(S$(SI),1)=" "
     S$(SI)=LEFT$(S$(SI),LEN(S$(SI))-1):
     GOTO 2010
2020 A$=RIGHT$(A$,LEN(A$)-J)
2030 IF LEFT$(A$,1)=" " A$=RIGHT$
     (A$,LEN(A$)-1): GOTO 2030
2040 SI=SI+1
2050 GOTO 1980
2060 S$(SI)=A$
2070 'SEARCH FOR COLUMNS
2080 FOR J=1 TO SI
2090 FOR I7=1 TO K
2100 IF S$(J)=L$(0,I7) GOTO 2150
2110 NEXT I7
2120 C(J)=VAL(S$(J))
2130 IF C(J)<0 OR C(J)>K THEN B$=S$(J):
     GOTO 1400
2140 GOTO 2160
2150 C(J)=I7
2160 NEXT J
2170 IF II<=1 GOTO 2330
2180 'SORT BY C(0)
2190 N(II+1)=N(2)
2200 I8=1
2210 I2=I8
2220 IF L$(N(I8+2),C(0))>=L$(N(I2),
     C(0)) GOTO 2280
2230 N(I2+1)=N(I2)
2240 I2=I2-1
2250 IF I2>0 GOTO 2220
2260 N(1)=N(I8+2)
2270 GOTO 2300
2280 'INSERT I8+2 HERE
2290 N(I2+1)=N(I8+2)
2300 'DONE
2310 I8=I8+1
2320 IF I8<II GOTO 2210
2330 J=1
2340 J1=60/SI
2350 PRINT " ID ";
2360 FOR I=1 TO SI
2370 PRINT TAB((I-1)*J1+5) L$(0,C(I));
2380 NEXT I
2390 PRINT
2400 IF II=0 GOTO 2500
2410 PRINT
2420 PRINT N(J);
2430 FOR I2=1 TO SI
2440 PRINT TAB((I2-1)*J1+5) L$(N(J),C(I2));
2450 NEXT I2
2460 J=J+1
2470 IF INT(J/12+.01)*12=J PRINT:
     INPUT "ENTER WHEN READY ";A$: GOTO 2350
2480 IF J<=II GOTO 2410
2490 PRINT
2500 PRINT
2510 GOTO 1200
2520 'CHANGE OR DELETE
2530 IF R<=0 GOTO 140
2540 CLS
2550 PRINT S$;
2560 PRINT TAB(4) "CHANGE OR DELETE"
2570 PRINT "ID NUMBER"
2580 PRINT
2590 FOR I=1 TO K
2600 PRINT I; " "; L$(0,I)
2610 NEXT
2620 I=0
```

```
2630 J1=192
2640 IF I=0 J1=148-KK
2650 GOSUB 3110
2660 IF B$="END" GOTO 140
2670 IF B$<>"" GOTO 2710
2680 I=I+1
2690 IF I<=K GOTO 2630
2700 GOTO 2620
2710 'SEARCH FOR B$
2720 IF I>0 GOTO 2760
2730 I8=VAL(B$)
2740 IF I8<1 OR I8>R GOTO 2620
2750 GOTO 2820
2760 I8=0
2770 I8=I8+1
2780 IF L$(I8,I)=B$ GOTO 2820
2790 IF I8<R GOTO 2770
2800 PRINT @ 908, E$; B$; " NOT FOUND"
2810 GOTO 2620
2820 'FOUND ENTRY AT I8
2830 J=0
2840 PRINT @ 82, I8;
2850 J=J+1
2860 PRINT @ 192+(J-1)*64+KK, L$(I8,J);
2870 IF J<K GOTO 2850
2880 'READY TO VERIFY
2890 PRINT @ 908, "CORRECT (Y/D/X) ";
2900 INPUT A$
2910 IF A$="Y" GOTO 2520
2920 IF ASC(A$)<>13 GOTO 2960
2930 IF I>0 GOTO 2770
2940 I8=I8+1
2950 GOTO 2740
2960 IF A$<>"D" GOTO 3040
2970 'DELETE ENTRY I8
2980 IF I8<R GOTO 2990
2990 FOR I=1 TO K
3000 L$(I8,I)=L$(R,I)
3010 NEXT
3020 R=R-1
3030 GOTO 2520
3040 I=VAL(A$)
3050 IF I=0 GOTO 2890
3060 'CHANGE ENTRY I
3070 J1=192
3080 GOSUB 3110
3090 L$(I8,I)=B$
3100 GOTO 2890
3110 'SUBROUTINE TO INPUT B$
3120 PRINT @ J1+(I-1)*64+KK,E$;
3130 PRINT CHR$(14);
3140 B$=""
3150 A$=INKEY$
3160 IF A$="" GOTO 3150
3170 IF ASC(A$)=13 PRINT CHR$(15);: RETURN
3180 IF ASC(A$)<>8 GOTO 3230
3190 IF LEN(B$)<1 GOTO 3150
3200 B$=LEFT$(B$,LEN(B$)-1)
3210 PRINT A$;
3220 GOTO 3150
3230 B$=B$+A$
3240 PRINT A$;
3250 GOTO 3150
```



Printing Pictures from Your Pet



BY MICHAEL D. GILLIE

Print Net gathers the characters off of the Commodore Pet's screen and prints them onto the Commodore 2022 tractor feed printer. A tractor feed printer is needed to eliminate gaps between the lines. This program is useful to make hardcopy graphic pictures. I developed this routine after unsuccessfully trying to take pictures of the screen using an inexpensive Polaroid. I now have the software equivalent of a camera (with a very slow exposure.)

To use Print Net as a subroutine change the END statement at line 60999 to RETURN. The routine uses less than 1200 bytes. You can take any Basic program and get a hard copy picture of it. Just insert a GOTO 60000 or a GOSUB 60000 (if you're making it into a subroutine) after the program draws a picture on the screen. Sometimes, if the program is large, you should use the CLR command (this is *not* the clear the screen function) before doing a GOTO or GOSUB. CLR erases all variables from memory, freeing up memory for the routine. CLR also erases all FOR/NEXT and GOSUB references, so never use CLR in a loop or after a GOSUB. You will need to Append (i.e., attach) Print Net to your program that draws the pictures. You can use either the Programmer's Toolkit by Palo Alto I.C.'s or a method by Brad Templeton which has appeared in the Pet User Notes (1-6).

If you don't have the Append capability and are copying your own pictures, a faster method would be to incorporate the Print Net routine into an existing drawing program.

Print Net can be added to the Mechanical Paintbrush program (February 1980 *Personal Computing*) or the "Draw" program by Peninsula School. You should consult the May issue of *Personal Computing* in the Feedback section for additions to the Mechanical Paintbrush program that allow you to save pictures on tape.

Listings 2 and 3 show how to incorporate Print Net into these programs. I have also written some changes that will allow Draw to work with both the old and new Pets.

When Print Net is added to either of these programs think of the total program as a merger of three routines: (1) Draw onto the screen; (2) Save or load tape; (3) Send to printer. *Warning:* When using either program, if you want to save the picture on tape you can't do it after you've sent it to the printer. When using this program, first save all your pictures on tape, creating an art portfolio. Whenever you want hardcopy just load the data tape (your art portfolio) and press SHIFT RETURN when you see a picture you want printed. SHIFT RETURN is CHR$(141) while RETURN is CHR$(13). I used the technique of assigning a function to a key. After you press the SHIFT RETURN keys, a net is woven over the screen and your picture is printed.

The accompanying pictures are mainly diversionary, but you can use the program for such things as letterheads, graphic illustrations, raffle tickets and Christmas cards. Let your imagination go. Another idea is to attach Print Net to a random art generating program; when you see a picture you like press SHIFT RETURN and you've got the picture on paper. Experiment and enjoy.

Programming Notes

Lines 60000 to 60200 convert the POKE values into a string using CHR$( ) and concatenate the string. See Table 1 for a list of variables and Table 2 for the POKE to CHR$( ) conversion scheme. Notice at line 60040, CF is set/reset to 0. That line and lines 60140 to 60170 insert RVS ON and RVS OFF control characters where needed. These routines minimize the number of RVS ON and RVS OFF control characters there are in a string.

If you build a string like: 10 FOR I=1 TO 40; A$=A$+"(RVS ON)X(RVS OFF)Y":NEXT I and you try to print it on the printer it won't come out. I don't know why this happens. I originally

Table 1

Variable	Purpose
A$( )	Lines of Print
R$	RVS ON Character
N$	RVS OFF Character (Normal)
I,J	Loop
L	Lower video address of line
U	Upper video address of line
CF	Control character Flag for reverse video
C$	Character added to A$( )
RF	Reverse Flag
D$	String of Data to create the special quote character
AN$	ANswer to an INPUT or GET

Mr. Gillie attends Lawrence Institute of Technology where he majors in Business Administration with a minor in Business Systems.

The first personal computer for under $200.

The Sinclair ZX80. A complete computer—only $199.95 *plus $5.00 shipping.*

Now, for just $199.95, you can get a complete, powerful, full-function computer, matching or surpassing other personal computers costing several times more.

It's the Sinclair ZX80, the computer that independent tests prove is faster than all previous personal computers. The computer that "Personal Computer World" gave 5 stars for 'excellent value.'

The ZX80 cuts away computer jargon and mystique. It takes you straight into BASIC, the most common, easy-to-use computer language.

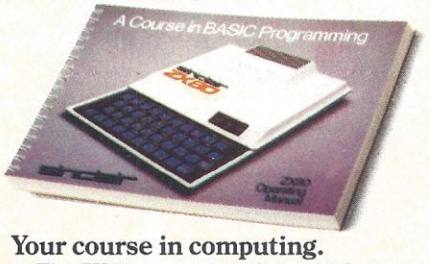
You simply take it out of the box, connect it to your TV, and turn it on. And if you want, you can use an ordinary cassette recorder to store programs. With the manual in your hand, you'll be running programs in an hour. Within a week, you'll be writing complex programs with confidence.

All for under $200.

Sophisticated design makes the ZX80 easy to learn, easy to use.

We've packed the conventional computer onto fewer, more powerful LSI chips—including the Z80A microprocessor, the faster version of the famous Z80. This makes the ZX80 the world's first truly portable computer (6½" x 8½" x 1½" and a mere 12 oz.). The ZX80 also features a touch sensitive, wipe-clean keyboard and a 32-character by 24-line display.

Yet, with all this power, the ZX80 is easy to use, even for beginners.



Your course in computing.

The ZX80 comes complete with its own 128-page guide to computing. The manual is perfect for both novice and expert. For every chapter of theory, there's a chapter of practice. So you learn by doing—not just by reading. It makes learning easy, exciting and enjoyable.

The ZX80's advanced design features.

Sinclair's 4K integer BASIC has performance features you'd expect only on much larger and more expensive computers. These include:

- Unique 'one touch' entry. Key words (RUN, PRINT, LIST, etc.) have their own single-key entry and are stored as a single character to reduce typing and save memory space.
- Automatic error detection. A cursor identifies errors immediately to prevent entering programs with faults.
- Powerful text editing facilities.
- Also programmable in machine code.
- Excellent string handling capability—up to 26 string variables of any length.
- Graphics, with 22 standard symbols.
- Built-in random number generator for games and simulations.

Sinclair's BASIC places no arbitrary restrictions on you—with many other flexible features, such as variable names of any length.

And the computer that can do so much for you now will do even more in the future. Options will include expansion of 1K user memory to 16K, a plug-in 8K floating-point BASIC chip, applications software, and other peripherals.

Order your ZX80 now!

The ZX80 is available only by mail from Sinclair, a leading manufacturer of consumer electronics worldwide. We've already sold tens of thousands of units in Europe, so demand will be great.

To order by mail, use the coupon below. But for fastest delivery, order by phone and charge to your Master Charge or VISA. The ZX80 is backed by a 30-day money-back guarantee, a 90-day limited warranty with a national service-by-mail facility, and extended service contracts are available for a minimal charge.

Price includes TV and cassette connectors, AC adaptor, and 128-page manual.

All you need to use your ZX80 is a standard TV (color or black and white). The ZX80 comes complete with connectors that easily hook up to the antenna terminals of your TV. Also included is a connector for a portable cassette recorder, if you choose to store programs. (You use an ordinary blank cassette.)

The ZX80 is a family learning aid. Children 10 and above will quickly understand the principles of computing—and have fun learning.

Phone orders only: (203) 265-9171. We'll refund the cost of your call.
Information: General and technical—(617) 367-1988, 367-1909, 367-1898, 367-2555. Phones open Monday-Friday from 8 AM to 8 PM EST.

sinclair

Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.

To: **Sinclair Research Ltd., 475 Main St., P.O. Box 3027, Wallingford, CT 06492.**

Please send me _____ ZX80 personal computer(s) at $199.95* each (US dollars), plus $5 shipping. (Your ZX80 may be tax deductible.)
I enclose a check/money order payable to Sinclair Research Ltd. for $_____.

Name _____

Address _____

City _____ State _____ Zip _____

Occupation: _____ Age: _____

Intended use of ZX80: _____

Have you ever used a computer? ☐ Yes ☐ No.
Do you own another personal computer? ☐ Yes ☐ No. *For Conn. deliveries, add sales tax.

PC-11-0

wanted to write this program with a TAB( ) function so I could put the picture anywhere on the paper, but because of this quirk the picture must go in the first 40 columns if it contains a string like the one in the above one-line code. If anyone knows a way to get around this or why it happens, drop a line to the Feedback section of this magazine.

Lines 60060 and 60070 convert the quote mark into a special character (CHR$(254) on the Commodore printers prints a self-designed character). In this case the special character looks exactly like a quote mark. If you use a quote mark in a drawing the printer will print the control characters after that (especially RVS control characters). You do not want this, you want what is on the screen; so use the special character for quotes.

Poke J, 91 in line 60180 prints the net onto the screen. This line shows that the computer is functioning. It can be replaced with an empty REM statement if you want the picture to remain intact.

Lines 60210 to 60999 do the actual printing. Line 60210 opens the necessary files. File 1 is for the printer, File 2 determines the line spacing and File 3 is for the special character. Files 2 and 3 are unique to the Commodore printers. Line 60220 defines the special character. Lines 60240, 60260, 60280 all define the spacing used. The formula is 144/# of lines per inch = value to be put in CHR$( ). For example CHR$(6) = 144/24, which means CHR$(6) sets spacing for 1/24 of an inch. You may have to fiddle with these lines and the print loops that follow them to get things perfect. The present values put 3 pictures on an 11-inch sheet.

Anything with [ ] means control characters (e.g. [3 LEFT] means 3 cursors left).

Line 60300 is a useful technique that has appeared in various magazines and user's notes. This line eliminates the problem of hitting RETURN with no data and thus exiting the program prematurely. If you have not seen this before, examine it and write a short program (under 5 lines) to see how it works. Every program for the Pet should use this technique or something like it. The format for a safe input is: Line # INPUT "Message [3 SPACES] Any single character you like [3 CURSOR LEFTS]; any string variable.

In my program if I hit RETURN without typing anything, AN$ will contain "?". Line 60310 just checks if the first character of AN$ is "N" for "no"; otherwise it will print the picture again. I don't like to type in answers so I get lines 60300 and 60310 to accept RETURN as a "yes" answer. You can change this by changing the question mark in line 60300 to an "N". Thus a "no" answer will become automatic.

For more information on the Programmer's Toolkit, contact your local dealer or Palo Alto I.C.'s, Division of Nestar Systems, 430 Sherman Ave., Palo Alto, CA 94306; (415) 327-0125. *Pet User Notes* is published by Commodore Business Machines, 3330 Scott Blvd., Santa Clara, CA 95051; (408) 727-1130. For more on "Draw," contact Peninsula School, Peninsula Way, Menlo Park, CA 94025. □

Program Listing 1- Print Net

```
60000 REM PRINT NET V1.5 M.GILLIE 4/80 FOR CBM 2022 & 8,16/32K CBM/PET
60010 DIM A$(24):R$=CHR$(18):N$=CHR$(146)
60020 FOR I=0 TO 24
60030   L=32768+I*40:U=L+39
60040   CF=0
60050   FOR J=L TO U
60060     IF PEEK(J)=34 THEN C$=CHR$(254):RF=0:GOTO 60140
60070     IF PEEK(J)=162 THEN C$=CHR$(254):RF=1:GOTO 60140
60080     IF PEEK(J)>=0 AND PEEK(J)<=63 THEN C$=CHR$(PEEK(J)+64):RF=0
60090     IF PEEK(J)>=64 AND PEEK(J)<=95 THEN C$=CHR$(PEEK(J)+128):RF=0
60100     IF PEEK(J)>=96 AND PEEK(J)<=127 THEN C$=CHR$(PEEK(J)+64):RF=0
60110     IF PEEK(J)>=128 AND PEEK(J)<=191 THEN C$=CHR$(PEEK(J)-64):RF=1
60120     IF PEEK(J)>=192 AND PEEK(J)<=223 THEN C$=CHR$(PEEK(J)):RF=1
60130     IF PEEK(J)>=224 AND PEEK(J)<=255 THEN C$=CHR$(PEEK(J)-64):RF=1
60140     IF CF=0 AND RF=0 THEN A$(I)=A$(I)+C$
60150     IF CF=0 AND RF=1 THEN CF=1:A$(I)=A$(I)+R$+C$:GOTO 60180
60160     IF CF=1 AND RF=0 THEN CF=0:A$(I)=A$(I)+N$+C$:GOTO 60180
60170     IF CF=1 AND RF=1 THEN A$(I)=A$(I)+C$
60180     POKE J,91
60190   NEXT J
60200 NEXT I
60210 PRINT "[CLR]":OPEN 1,4:OPEN 2,4,6:OPEN 3,4,5
60220 D$=CHR$(0)+CHR$(112)+CHR$(0)+CHR$(112)+CHR$(0)+CHR$(0):PRINT#3,D$
60230 PRINT#1,"[HOME]"
60240 PRINT#2,CHR$(6)
60250 FOR I=1 TO 11:PRINT#1:NEXT I
60260 PRINT#2,CHR$(18)
60270 FOR I=0 TO 24:PRINT#1,A$(I):NEXT I
60280 PRINT#2,CHR$(6)
60290 FOR I=1 TO 11:PRINT#1:NEXT I
60300 INPUT "AGAIN   ?[ 3 LEFT]";AN$
60310 IF LEFT$(AN$,1)<>"N" GOTO 60220
60320 PRINT#2,CHR$(24)
60330 CLOSE 1:CLOSE 2:CLOSE 3
60999 END
```

Table 2

Characters	J=POKE	CHR$
ASCII	0-63	J+64
Shifted ASCII	64-95	J+128
Rest of graphics	96-127	J+64
RVS ASCII	128-191	J−64
RVS Shifted ASCII	192-223	same
RVS Rest of graphics	225-255	J−64

Program Listing 2

Changes after Mechanical Paintbrush and Print Net have been merged

```
    6 P$=CHR$(141):REM SHIFT RETURN
  103 IF A$=P$ GOTO 60000
  263 IF E$=P$ GOTO 60000
60999 CLR:GOTO 1
```

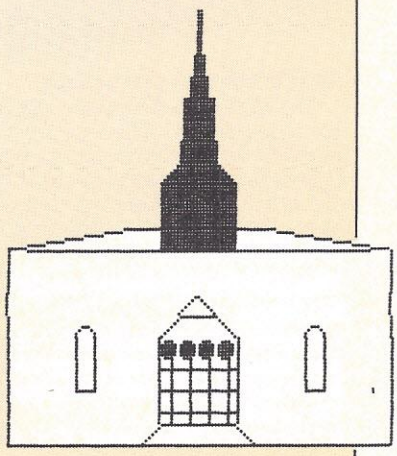
Program Listing 3

Changes after Draw and Print Net have been merged

```
  123 IF C$=CHR$(141) GOTO 60000
60999 CLR:GOTO 1
```

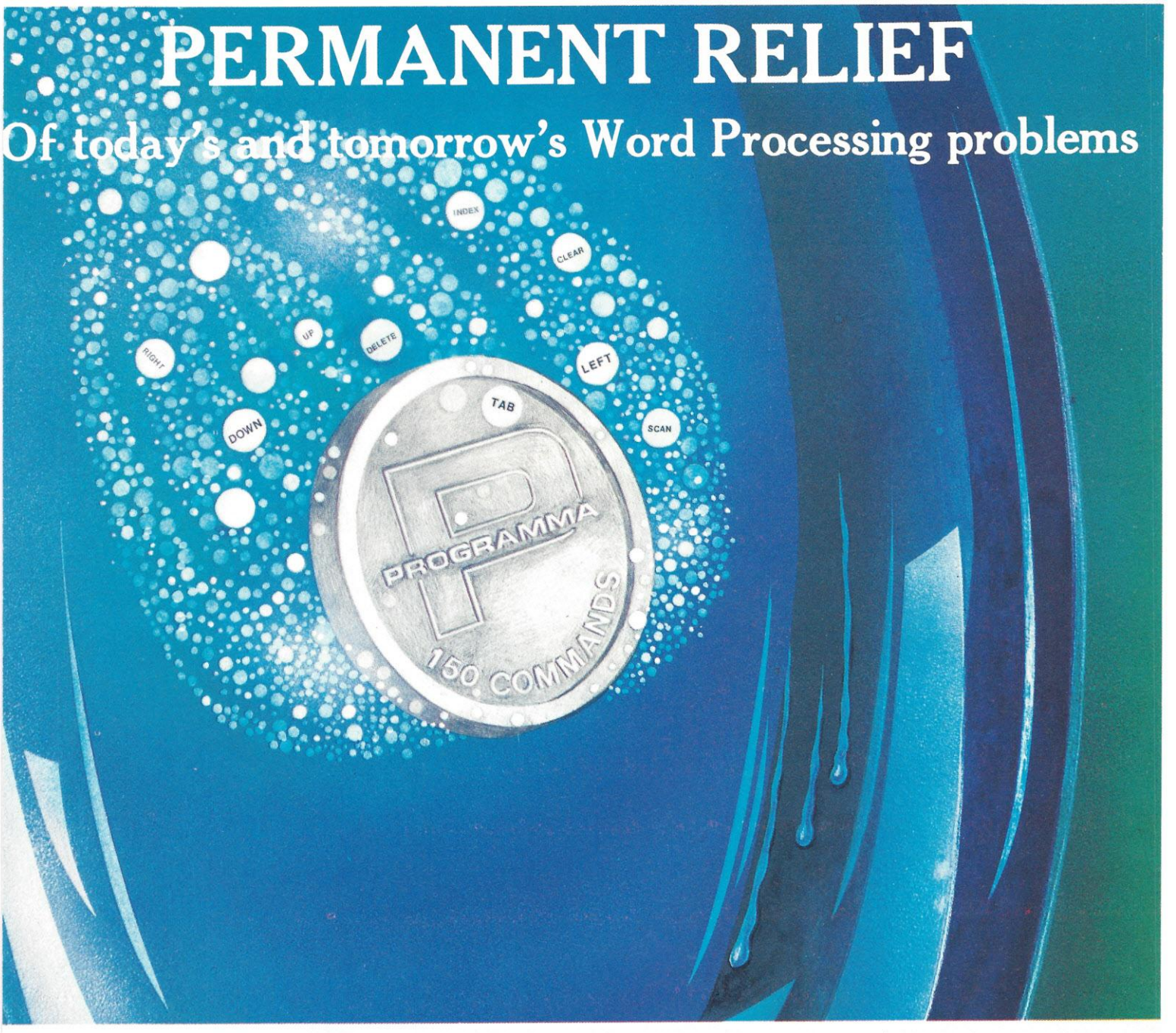
FOR NEW ROM

```
   27 IF PEEK(12*4096)<>0 THEN IB=158:KD=151
 5673 IF PEEK(12*4096)<>0 GOTO 5678
```



PERMANENT RELIEF

Of today's and tomorrow's Word Processing problems



Apple PIE + Formatter

Apple PIE (Programma International Editor) and FORMAT (text formatter) offer full strength solutions to today's word processing problems. These versatile, powerful programs provide document preparation and word processing capabilities previously found only on much larger computer systems.

PIE is a general purpose, full screen editor that uses control keys and function buttons to provide a full range of editing capabilities such as search and replace, delete, copy, insert, move. Changes may be made directly anywhere on the screen and are shown as they are performed.

FORMAT uses simple instructions embedded in the input text to describe the desired appearance of the final document. It handles centering, underlining, indenting, page numbering, margins, headers, footers, even form letters, and includes a proofing capability.

These high-quality, cost-effective programs come with comprehensive documentation and run on a 32K Apple II. They are available through your local computer store or direct from Programma International, Inc. at the introductory price of $79.95*.

VIDEX VERSION T.M.
DOUBLE VISION T.M.
SUPR TERM VERSION T.M.
STANDARD VERSION

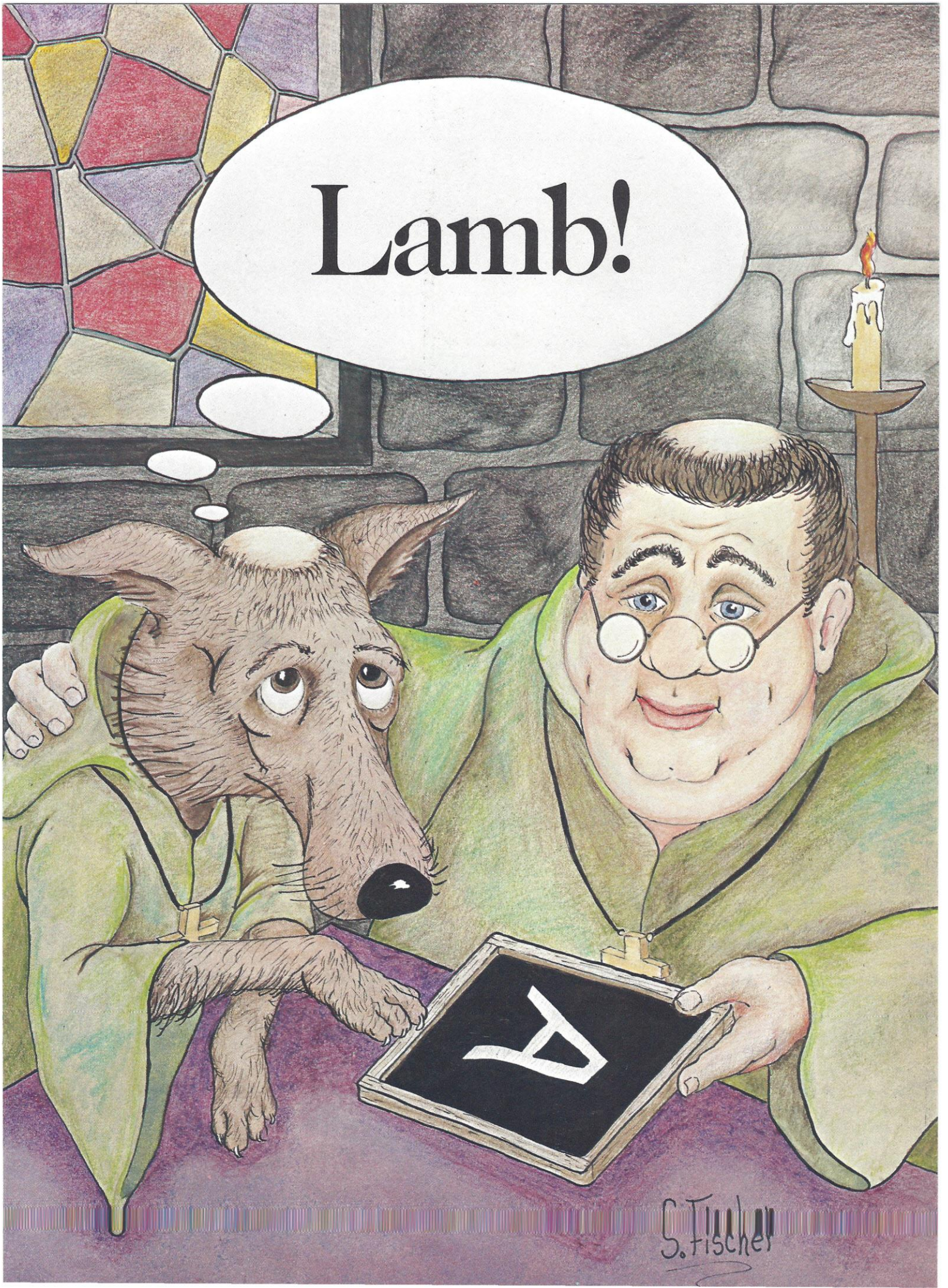
*December 1, $129.95.

PROGRAMMA
3400 Wilshire Boulevard
Los Angeles, California 90010

or

CIRCLE 19

Simple enough for the beginner. Versatile enough for the professional.


Lamb!
A
S. Fischer

BY DAVID GERROLD

All right, I confess. I really bought the computer for the games.

Yeah, sure — I do all the other stuff with it. I use it as a word processor, and for counting calories, and making a budget and keeping tax records, and storing phone numbers, and organizing my notes and files, and predicting biorhythms and impressing hell out of my friends and family — but the *real* reason I bought the computer is that I wanted to have *fun* with it. I wanted to play Chess, and Adventure, and Othello, and Lunar Lander. (For some reason, I have no great urge to play Star Trek . . .but I suspect that this is a personal quirk, and not a judgmental decision.)

Having finally gotten that off my chest, there are three observations I would like to make about buying a computer to play games.

1. You're a long way from the *lamb*.

Let me tell this as a fable:

Once upon a time there was a wolf who heard that the brothers of a certain order always ate lamb. Having more than a passing fondness for lamb himself, the wolf presented himself at the monastery and asked if he could join. The monks explained to him that it took a lot of hard work and study to join their order. Their purpose was to copy the holy scrolls and study them, so they could discuss the insights inscribed therein and thereby gain new perceptions of the glory of God's universe. The wolf said he was willing to work very hard, just one question — this was the order where they ate lamb, wasn't it?

The brothers of the order took the wolf in — with some reservations, of course; but then again, who were they to keep a seeker of truth (of *any* species) from his quest? — and one of their number was assigned to tutor him. The brother began by drawing an A on his slate. "That's a lamb, right?" said the wolf.

"No, it's an A."

"Oh."

It took the poor wolf a week to learn this fact, but he was determined to succeed — so he could eat lamb — and after a great deal of effort, he learned to recognize the A.

The patient monk now drew another letter on his slate. "That's a lamb, right?" asked the wolf.

"Uh — no, it's a B."

"Oh."

This time, it took the poor wolf two weeks to learn to recognize the letter, and he worked very hard at it, so much so that he actually began to feel a little cocky. So when his tutor drew a third letter on the slate and said, "Now, this is a — " the wolf shouted, "*Lamb!* Let's eat!"

That's when they threw him out.

Those who get into computing for the fun of writing and playing games are very much like that poor wolf. We have to learn our Basics first.

And usually, Basic isn't enough.

You very quickly discover that Basic is lousy for real-time simulations like Space War and Lunar Lander, and it's dreadfully slow for just about everything that has nested loops. If the machine takes more than fifteen seconds to return a response, your friends aren't impressed at all — instead, they say something like, "Gee, Howard, that's great. Uh — how much did you say all this stuff cost you?"

"". . .who were they to keep a Seeker of truth from his quest?"

2. The software is always greener on the other guy's system.

I don't know a programmer who doesn't have this complaint.

If you see a program you like, you want it.

But it is almost always impossible for you to obtain.

It takes more memory than you have. *Or* —

It's written in a language you don't have. *Or* —

It's only available for the TRS-80, and you have an Apple (or vice versa). *Or* —

All you have is a listing, and it's in a dialect of Basic so bizarre that not even your friend who does binary arithmetic in his head knows how to translate it. *Or* —

You get the idea.

There's all that terrific software available. You just can't use it on your system. (Does anyone know where I can get Adventure on a North Star disk?)

3. Those games that you *can* get into your system are almost always obsolete.

I suppose part of this is due to the fact that the amount of memory available to the average user has grown enormously. Not too long ago, 8K was considered to be a lot of memory. Now, 16K is barely enough.

But we have all those games that were written simple because there wasn't the space to do more.

(And one must also remember that many of those early games were, in one sense, pioneering efforts in a new entertainment medium — a medium that is growing so fast that it hasn't even been recognized as the phenomenon it is yet.)

It is only since larger memories have become widely available that more sophisticated simulations have started to appear on a wide scale. Like Adventure, for instance.

Now, I have some ideas for some games that I would like to play — but nobody has written them yet.

And I know where that leads. A few years back, I had some ideas for some science fiction stories I wanted to read, but couldn't find them on the library shelves, so I had to write them myself.

I have this dreadful sinking feeling that I am going to have to write these games for myself as well.

And that brings me back to 1. It's still a long way to the *lamb*.

But it's going to be worth it. If I have the patience. All right, hold the slate steady. That's a C, right? □

Gerrold has published over 20 books, including five anthologies, a short story collection, ten novels (including When Harlie Was One *and* The Man Who Folded Himself*), and two non-fiction books about "Star Trek":* The World of Star Trek *and* The Trouble With Tribbles. *Gerrold has done a number of other TV shows, including "Land of the Lost" and "Logan's Run." He also has a monthly column about writing in* Starlog.

Illustration by Stephen C. Fischer

Election Night

BY NANCY G. McPHEE

Winston Churchill said, "Democracy is the worst possible form of government — except for all the others." In a democratic country, elections are almost as inevitable as death and taxes, and November will see national, state and local elections across the United States. On election night candidates' committee rooms will be a brou-ha-ha of ringing phones, clacking adding machines, blaring TV sets, chalk screeching on blackboards, cheers and moans as workers scramble to keep up with the results.

Here's a program to bring order out of all that chaos, to compile, display and interpret local election figures just as the TV network computers do for the national results. If you're a political buff, as I am, you can do your favorite candidate a good turn by setting up shop in his headquarters. If you're not, you might well be able to interest a local campaign manager in purchasing your services for the evening. And even if you wouldn't be caught dead in the vicinity of a politician, you may be interested in some of the program's techniques — including a nifty routine for coding tape data which reduces the normal time and tape storage requirements by 95%! Best of all, Election Night has been tested under fire in the recent Canadian federal election.

U.S. elections have at least one merit — they occur at predictable intervals. In Canada, where politics has become something of a national sport, governments can be defeated and elections take place any old time. Ours has always been a political family, on the principle that it's better to try to change the things you don't like than to sit around and complain. My husband John has been a candidate himself and a stout campaigner for others; and our three youngsters are under the impression that everyone grows up stuffing envelopes and banging signs into front lawns. I'm a back-room girl myself, and have managed several campaigns.

Having recently acquired a TRS-80 (named Hermie) we've all become computer freaks as well, and the new enthusiasm threatens to compete with the old. When a federal election was called for February 1980, hard on the heels of one in June 1979, it did seem like too much of a good thing, particularly as the campaign could take us away from our beloved Hermie. So while John set out on a frosty round of door-knocking and subway-stopping with the candidate, and our teen-aged boys ruefully pondered the possibilities of pounding wooden sign-stakes into concrete-frozen ground, I sneakily cast about for some way of doing my bit while still playing with the computer. The result was Election Night.

The program displays and constantly updates the running totals for all the candidates in an electoral area — be it parliamentary riding, congressional district, city ward or whatever — keeps track of popular vote percentages and the percentage of the vote reported, displays the results in any individual polling subdivision or precinct and compares them with the results from a previous election. Even when new results are being entered the updated totals and percentages are always on the screen. I've always hated committee rooms on election nights because in spite of all the hyperactivity it's hard to form any pattern of what's going on. With this program, the strategists can see at a glance how many polls have reported, check key or bellweather precincts, analyze subtle shifts in the vote percentages and interpret the trends. Party workers can instantly see the results for the area they

---

Ms. McPhee is the author of The Book of Insults *($2.95), published by Van Nostrand Reinhold Ltd., Toronto. Her lifelong fascination with politics led her as a volunteer into the back rooms, and she was a campaign manager in several successful federal and provincial election campaigns.*

canvassed, compare them with last time, and tell whether they have made up, or retained, the necessary votes. Who needs NBC?

Election Night is a simple and straight-forward program. Written for a TRS-80 in standard Microsoft Basic, and using very few special TRS-80 functions, it should be easily adaptable to other Basics with slight modifications. The program occupies just under 5K bytes of memory and runs in about 13K. It uses cassette storage for the results of the previous election, if you have them. While it can easily be adapted to disk storage, the tape coding technique used is so efficient that it doesn't make sense to cart delicate disk drives into the traditionally smoke-filled rooms.

A description of variable names accompanies the listing, as does the listing for POLLDATA, a short program to create the data tapes. I have used the actual candidates' names and party labels from our constituency of Etobicoke Centre. The program was written to accommodate four candidates, which by coincidence just fills the screen nicely. The party labels LIB, PC and NDP stand for Liberal, Progressive Conservative and New Democratic Party, while OTHERS combined a gaggle of fringe candidates of varying lunatic views. (My favorite was the Rhinoceros Party, named for the stupidest and thickest-skinned beast around, whose main campaign promise was not to keep any of its campaign promises. It was a real temptation to vote Rhino!) If you have fewer candidates, you will need to insert blank lines here and there in the display. In any case, you'll have to change names and labels to fit your local situation, and play around with TAB settings depending on the length of the names.

Tips For Using Election Night

As with most programs of this kind, data entry is the bugbear. When the results start coming in, they come in a flood. As the slips of paper mount up and you feel the beery breath of the party faithful on your neck, it's easy to get rattled and make mistakes. The program has traps for the most obvious entry errors, and a routine to permit you to change incorrect entries, but in the heat of battle it's still possible to foul things up. Ask the one who did it!

If you're planning to use Election Night in real-time, efficient planning, practice and discipline are the keys. Be sure to check the order (usually alphabetical) in which results will be reported and structure the input section of the program accordingly. Have "in" and "out" baskets strategically placed so that it's clear which polls have already been entered. Establish a routine and practice the data entry procedures in advance so that you are thoroughly familiar with them and train someone else, if you can, so that you can take frequent breaks. Above all, keep calm and don't feel rushed. This is particularly important if you have an emotional involvement in the outcome. (My dispassionate 14-year-old, Patrick, was much more efficient at entering results than I was, and ultimately took over entirely!)

The INKEY$ routine for selecting program functions permits fast data entry, but it has its pitfalls. Since no prompts are displayed, it's easy to forget the proper entry or miss a step. A backspace will display an individual poll and a "*" will let you change poll figures. Entering a new poll requires that ENTER (or any other key) be pressed and then the poll number be entered. In my hurry I tended at first to forget ENTER and just type in the poll number, resulting in occassional inaccuracies and the necessity for corrections. My other problem was that often the poll slips did not bother to show a figure for OTHERS and I tended to press ENTER instead of 0. This added the most recent value of the V4 variable to the OTHERS total, and as time went on it became obvious that this total was inaccurate. By checking the individual polls I was able to track down the errors and correct the totals, but this was both time-consuming and bad for the image. So be careful, enter a 0 if there is no recorded vote for a candidate. Again, practice and a disciplined procedure are the keys.

Originally we had planned to route the results slips to the computer first and then to the wall-boards for manual entry. We quickly found, however, that a single terminal could not keep up with a battery of adding machines and half-a-dozen runners, and made the computer the final stop. Even so, as the evening wore on the bulk of the crowd gathered round the computer, where the pattern of the results could be seen at a glance.

In one way, the program was almost too efficient. Although our candidate was leading in the totals well into the evening, the computer told those in the know a different story. By checking key polls and comparing their results with 1979, we knew we had fallen short and weren't going to make it. In the event we lost by several thousand votes. Still, if our candidate lost, our party won overall, and Election Night itself proved to be a resounding success.

Notes and Comments on Election Night

The program is written in six segments. Lines 10-220 form an initialization sequence; lines 300-440 display the vote totals and percentages; lines 500-570 set up an invisible INKEY$ routine to select the program's other functions. Lines 600-750 enter and total the results; lines 800-960 display the results for an individual poll; and lines 1000-1260 permit the figures for a poll to be corrected. Lines 1300-1500 provide an optional routine to input and decode previously recorded data from tape.

One note: my printer types ")" for "greater than" and "(" for "less than". In context you should have no trouble distinguishing these from parentheses.

Lines 10-210: Initialization Sequence

Lines 60 and 1620-1630 provide (in theory) an error-trapping routine which prevents a program crash during operation by printing the TRS-80 error code and line number and then returning to the main program. In practice I have found the ON ERROR GOTO function rather erratic, so don't worry if you don't have it, just omit these lines. In any case, if you encounter an error before line 300 it's best just to restart the program. However, if the program crashes for any reason after there is data entered into memory, *do not* restart it with RUN. This would destroy all the results you have so far. Instead, type GOTO 300, which will re-enter the main program without initializing the variables. Of course, the obvious procedure is to test each section thoroughly before using the program in earnest, to eliminate those typing and punctuation errors that will creep in.

Line 70 clears the screen and clears 500 bytes of string space. The program itself uses very little string space, since most of the text printed on the screen is in the form of quoted string constants; 200 bytes is more than ample for miscellaneous string storage. However, the tape decoding routine, if you decide to use it, requires about an additional 300 bytes.

Line 80 defines almost all variables as integers and sets up two arrays: POLL, which will store the results to be entered during the evening, and LAST, which stores the results of the most recent election for comparative purposes. The size of these arrays is directly related to the number of polls or precincts you wish to cover and the number of pieces of data

you want to store for each poll. In my case, there were 303 polls, and I wanted the vote for each of four candidates plus the total vote for the poll. Hence, POLL (303,5). Change the dimensions to suit your own requirements.

The size of these arrays dramatically affects the amount of memory required to run the program. As line 80 is written, the two arrays use 7780 bytes of memory in addition to the 4913 bytes occupied by the program itself; reducing the number of polls to 200 would reduce the necessary memory by about a third and increasing the number would increase the memory requirement correspondingly. If you don't have or don't want to use any previous election results, eliminating the LAST array cuts the memory needed in half. You can experiment with the size of your arrays by running the program until the first prompt appears; then press BREAK and print the available memory. If it is uncomfortably small, you may have to reduce your array dimensions.

It was this consideration for memory which led me to define most of the variables as integers. The program was first written with all variables as single-precision (the TRS-80 default value) and needed nearly 20K to run; the arrays occupied 15068 bytes. As integers, they are stored in a much more compact form and the array memory requirement is reduced by almost half, permitting the program to run comfortably in 16K and making it unnecessary for me to lug my expansion interface out into a cold, cruel world. DEFINT is a TRS-80 statement defining the indicated variables as integers. It can be overidden by a type declaration character such as "$" for string or "!" for single-precision. Since the only variables likely to exceed 32767 are those for the various total votes, these (LT, PT, NT, OT and TV) have to be followed by "!"; this character becomes part of the variable name and must be used each time.

Lines 100 to 150 set up the initial vote totals; normally there won't be any, and you can just pres ENTER to the prompts, but this feature permits you to start (or restart) the program even if some partial results are in. It might not be a bad idea to keep rough track of the totals on a scratchpad during the evening, just in case some gremlin at the power company pulls the plug. Line 160 is a fancy touch. The initial value for TV! would normally be 0, and several later lines use TV! as a divisor. Adding 1 at this point permitted me to check out the various program functions before the polls closed without encountering an obnoxious "division by zero" error message. To keep things honest, this spurious extra vote is subtracted in line 720 with the first poll entry, so the total

PRINT USING

Formatting Strings

PRINT USING is an extremely flexible and useful function, but at times a confusing one. The Level II manual goes on and on about it, and I for one found myself skipping over the detail ("I'll never remember all that.") As with many complex functions, it's best to learn a bit at a time, using only what you need and postponing the rest.

PRINT USING prints numerical information in a uniform and pre-determined form. A "formatting string" determines how many digits will be printed, how many decimal places and whether commas will be inserted at the appropriate places. My formatting strings are defined in line 330. Figures printed using A$ will be whole numbers containing up to 5 digits with a comma inserted in any number over 999. I used A$ for all the running vote totals, none of which would exceed 99,000. Those using B$ will contain 4 digits and will be rounded to one decimal place; B$ was used for all percentages. C$ will produce a 3-digit whole number, and was used for individual poll entries and totals. If the number does not have enough digits to fill the allotted format, PRINT USING prints leading blanks as place holders. The effect of printing a series of numbers at the same tab setting using the same formatting string is to stack them in neat columns with decimal points and commas aligned.

Punctuation

PRINT USING has some tricky punctuation requirements which are not addressed in the Level II manual. First a review.

It is possible to use

```
PRINT TAB(A) X; TAB(B) Y$; TAB(C) "JOE SMITH";
                TAB(D) Z
```

This is all one statement; PRINT does not have to be repeated, and the semi-colons prevent a carriage return between the items to be printed. Some Basics also require a semi-colon after the tab setting:

```
PRINT TAB(A); X; TAB(B); Y$;...etc.
```

In Radio Shack Basic this is optional, but for reasons about to become apparent it is probably a good idea to acquire the habit of always using this semi-colon.

The following will also run successfully:

```
PRINT TAB(A) X; TAB(B); Y$; TAB(C); USING A$; Z
```

but in this case, the semi-colons following TAB(C) and A$ are mandatory. PRINT USING requires a semi-colon after any tab setting and after the name of the formatting string.

The fun begins when you attempt to combine a print series containing several PRINT USING statements. Line 380 of Election Night originally read:

```
380 PRINT TAB(14) "CRUDEN (LIB)"; TAB(34); USING
    A$: LT!; TAB(44); "("; USING B$ LT!/TV!*100; "%)"
```

Each time I ran it, the program crashed at TAB(44) with the complaint "Type mismatch error." What nonsense! Stupid machine! Must be a ROM error!

An hour's hair-pulling, teeth-gnashing and laborious trial and error produced the solution. The line had to be written:

```
380 PRINT TAB(14) "CRUDEN (LIB); TAB(34); USING
    A$; LT!; PRINT TAB(44); "("; USING B$; LT!/TV
    !*100;: PRINT "%)"
```

The Rule: After PRINT USING and a numerical variable it is necessary to insert :PRINT before a string constant or variable — in effect, to start a new PRINT statement.

The Reason: After a PRINT USING statement the computer is looking for a numeric value to put into string form. If it encounters a string, as it did with my ")", the poor thing hiccups and cries "Type mismatch"!

— by Nancy G. McPhee

vote is accurate as soon as the actual results start rolling in.

Lines 170 to 200 permit the selection of subroutines to input taped data with the results of a previous election, and to correct any errors in that data. Delete them if you are not using any previous results, and with them lines 1300-1500. (At the same time, change the 500 in line 70 to 300, and remove the LAST array from line 80.) In any case, read the comments later in this description on the tape-input routine — it's an affecting tale and you may be able to use this method in some of your other programs. The variable F in line 200 is a flag; the same subroutine is also called from elsewhere in the program, and F=1 lets the TRS-80 know that in this case it is the LAST array which is to be changed.

Lines 300-440: Display and Summary of Vote Totals

This routine is the core of Election Night. It produces a display which constantly occupies the top half of the screen, showing the number of polls reporting and the percentage of the total; the current total vote for each candidate and his percentage of the popular vote; and the total votes recorded so far. As each new poll is entered in the bottom half of the screen, the totals are updated, the screen clears for a split-second and then the new totals are displayed.

Formatting a screen display presents some finicky but not really difficult problems. For me it was largely a question of trial and error, playing around with the best TAB settings to give a pleasing visual effect. I found the TRS-80 PRINT USING function invaluable, and learned the hard way a few tricks about its use which are summarized elsewhere. Suffice it to say here that the punctuation in this routine and in the subroutine at lines 800-960 is absolutely critical. Leave out a semi-colon and you're dead! Note in these routines that a semi-colon at the end of a line suppresses the line-feed so the following line prints as part of the same screen line. If your Basic doesn't have PRINT USING or a similar function for printing figures in tidy columns you'll have to work out your own solution to a neat display.

Line 330 sets up three "formatting" strings, used by PRINT USING to round up figures and print them in a consistent format. I've covered the definition of these strings a little more fully elsewhere. Lines 350-360 print the number of polls currently reporting and calculate the percentage of the total.

Line 370 illustrates a neat way of printing a multiple of a single character — in this case a line of "+" across the screen. Note that only 63 are printed, although there are 64 columns on the screen; printing 64 characters moves the cursor to the next line before the linefeed, and the effect is to leave an unwanted blank line on the screen. Lines 380-410 print the current totals and percentages for each candidate. I set this up with our own candidate at the top and the most dangerous opposition (alas, the ultimate victor) next. Line 420 prints the total vote so far and line 430 encloses the display with another line of crosses.

Lines 500-570: INKEY$ Routine to Choose Functions

After printing the top half of the screen, the program immediately proceeds to a routine permitting a choice of functions: you may enter results for a new poll, display the results for any individual poll or change the figures for any poll which may have been entered incorrectly. The routine uses INKEY$, a TRS-80 function which strobes the keyboard looking for a depressed key. It's similar to the GET function on the Pet and Apple. Because it doesn't print any prompt or scroll the display, it keeps the screen display clean. If you don't have it, INPUT will do (although in my case it would have spoiled the top half of the display.) INKEY$ has its minor pitfalls during program operation, as I have noted.

Line 530 defines X$ as whatever key is depressed; if there is no response it loops to itself until it detects an entry. In line 540, if the key pressed is an asterisk, the branch is to the subroutine to change poll figures. Any character could be used; I chose an asterisk because it was unlikely to be pressed by accident. The flag variable F is set to 2, again to indicate to the subroutine which section of the program is calling it. Line 550 looks for a backspace (CHR$(8)), another key unlikely to be pressed in error. If it finds it, it goes to the subroutine to display individual polls. Any other key, including ENTER, will send the program to the subroutine to enter and total new results. All the subroutines return to line 560, which loops back to the top of the main display.

Lines 600-750: Enter and Total Poll Results

This very simple subroutine is the program's workhorse. It fills the bottom of the screen while the running totals remain on top, and performs the entry of each poll's results and the updating of all the totals.

Line 640 inputs the poll number; one of the few error-traps in the program is the second half of the line, which will not permit a poll number higher than the dimensions of the array. Lines 650-680 input the results for each candidate. Note that the order of entry has been switched to correspond to the alphabetical listing on the ballot. Line 690 totals the votes. Variables V1, V2, V3, V4 and VT are temporary variables; they do not store the results, but change value with each new poll. Lines 700-710 store the results in the appropriate row and column of the POLL array, add the poll total to the total vote and increment the number of polls reporting by 1. If this is the first poll reporting, line 720 deducts the false vote entered in line 160; this will only happen once. Line 730 adds each candidate's vote in this poll to his total vote. The subroutine then returns and the top of the screen is reprinted with the fresh totals and percentages.

Lines 800-960: Display Individual Poll Results

This subroutine is similar to the routine to display running totals and should be easy to follow. Like the other subroutines, it fills the bottom of the screen while the running totals remain on top. It displays the poll number, the current figures on the left side of the screen and the comparative figures for the most recent election on the right side. I used the candidate's names for the current election and party labels only for the previous figures. If you aren't using previous figures, you will want to center the current year's display. Line 940 allows this display to remain on the screen as long as desired; ENTER will clear the screen and return to the main routine.

Lines 1000-1260: Change Poll Data

This subroutine is not strictly necessary. It permits any erroneous entries to be corrected, and thest can easily happen. It is almost identical in structure to the poll entry routine except that it uses a couple of internal subroutines. Lines 1050-1060 input the poll number and check that it is not out of range. Line 1070 checks the flag variable; if F=2 the routine is updating current poll data, so it branches to line 1170, which deducts the figures for this poll from the running totals. Lines 1080-1120 input and total the new figures. Lines 1130-1140 check the flag again; if F=1, the branch is to line 1200, which changes the figures in LAST, if F=2, the branch is to lines 1230-1240, which change the POLL array and add back the corrected figures to the running totals.

continued on p. 64

Election Night Program Listing

```
10  ' "ELECTION NIGHT" -- A PROGRAM TO TOTAL AND DISPLAY
      ELECTION RESULTS
20  ' WRITTEN BY NANCY G. McPHEE      FEBRUARY 1980
30  '
40  '*********************************************************
50  '
60  ON ERROR GOTO 1620
70  CLS: CLEAR 500
80  DEFINT F-X: DIM POLL(303,5), LAST(303,5)
90  PRINT "ELECTION RESULT PROGRAM -- INITIALIZATION SEQUENCE"
100 PRINT: INPUT "WHAT IS LIBERAL TOTAL"; LT!
110 INPUT "WHAT IS PC TOTAL"; PT!
120 INPUT "WHAT IS NDP TOTAL"; NT!
130 INPUT "WHAT IS 'OTHERS' TOTAL"; OT!
140 INPUT "HOW MANY POLLS REPORTING"; NP
150 TV! = LT! + PT! + NT! + OT! : PRINT "TOTAL VOTE IS "; TV!
160 IF TV! = 0 THEN TV! = 1
170 INPUT "DO YOU WANT TO INPUT RECORDED POLL DATA (1/0)"; X
180 IF X = 1 THEN GOSUB 1300 ELSE 300
190 INPUT "DO YOU WANT TO CHANGE POLL DATA (1/0)"; X
200 IF X = 1 THEN  F = 1: GOSUB 1000: ELSE 300
210 '
300 ' *********************************************************
310 'ROUTINE TO PRINT RUNNING TOTAL OF VOTE ON SCREEN
320 '
330 A$ = "##,###": B$ = "###.#" : C$ = "###"
340 CLS: PRINT TAB(14) "ETOBICOKE CENTRE -- ELECTION RESULTS"
350 PRINT TAB(4) "NUMBER OF POLLS REPORTING: "; USING C$; NP;
360 PRINT " / 303  ("; USING B$; NP/303*100; : PRINT
     " % OF TOTAL)"
370 PRINT STRING$(63,"+")
380 PRINT TAB(14) "CRUDEN  (LIB)"; TAB(34); USING A$; LT!;
    : PRINT TAB(44); "("; USING B$; LT!/TV!*100;: PRINT " %)"
390 PRINT TAB(14) "WILSON  (PC)"; TAB(34); USING A$; PT!;
    : PRINT TAB(44); "("; USING B$; PT!/TV!*100;: PRINT " %)"
400 PRINT TAB(14) "SHIPLEY (NDP)"; TAB(34); USING A$; NT!;
    : PRINT TAB(44); "("; USING B$; NT!/TV!*100;: PRINT " %)"
410 PRINT TAB(14) "OTHERS"; TAB(34); USING A$; OT!;
    : PRINT TAB(44); "("; USING B$; OT!/TV!*100;: PRINT " %)"
420 PRINT TAB(14) "TOTAL VOTE:"; TAB(34); USING A$; TV!
430 PRINT STRING$(63,"+")
440 '
500 ' *********************************************************
510 ' ROUTINE TO CHOOSE FUNCTIONS: ENTER, DISPLAY OR CHANGE
      INDIVIDUAL POLLS
520 '
530 X$ = INKEY$ : IF X$ = "" THEN 530
540 IF X$ = "*" THEN F = 2: GOSUB 1000: GOTO 560      'CHANGE
550 IF X$ = CHR$(8) THEN GOSUB 800 ELSE GOSUB 600
    'DISPLAY / ENTER
560 GOTO 340
1160 '
1170 LT! = LT! - POLL(I,1): PT! = PT! - POLL(I,2): NT! =
     NT! - POLL(I,3): OT! = OT! - POLL(I,4): TV! = TV! -
     POLL(I,5)
1180 RETURN
1190 '
1200 LAST(I,1) = V1: LAST(I,2) = V2: LAST(I,3) = V3:
     LAST(I,4) = V4: LAST(I,5) = VT
1210 RETURN
1220 '
1230 POLL(I,1) = V1: POLL(I,2) = V2: POLL(I,3) = V3:
     POLL(I,4) = V4: POLL(I,5) =  VT
1240 LT! = LT!+V1: PT! = PT!+V2: NT! = NT!+V3: OT! = OT!+V4:
     TV! = TV!+VT
1250 RETURN
1260 '
1300 ' *********************************************************
1310 'SUBROUTINE TO INPUT AND DECODE DATA FROM TAPE
1320 '
1330 INPUT "(ENTER) WHEN RECORDER READY TO INPUT DATA"; X
1340 INPUT "WHAT IS FILENAME"; F$
1350 INPUT #-1, F1$, R1, R2
1360 PRINT "FOUND "; F1$; "; POLL"; R1; " TO"; R2
1370 INPUT "IS THIS CORRECT (1/0)"; X: IF X=0 THEN 1330
1380 I = R1: J = 1: R$ = "": T$ = "/": PRINT I;
1390 INPUT #-1, R$: SP = 1
1400 FOR K = 1 TO LEN(R$)
1410    IF MID$(R$,K,1) = T$ THEN NC = K-SP ELSE 1450
1420    LAST(I,J) = VAL(MID$(R$,SP,NC))
1430    SP = K+1: J = J+1: IF J ) 5 THEN J=1: I = I+1:
        PRINT I;
1440    IF I ) R2 THEN 1460
1450 NEXT K: R$ = "": GOTO 1390
1460 PRINT: PRINT "DATA IS IN MEMORY"
1470 INPUT "DO YOU HAVE MORE DATA ON TAPE (1/0)"; X
1480 IF X=1 THEN 1330
1490 RETURN
1500 '
1600 ' *********************************************************
1610 '
1620 PRINT "ERROR "; ERR/2+1; "IN LINE "; ERL
1630 RESUME 340
1640 END
```

Polldata Program Listing

```
10  ' "POLLDATA" -- A PROGRAM TO ENCODE AND RECORD POLL FIGURES
      ON TAPE FOR USE IN "ELECTION NIGHT"
20  '  WRITTEN BY NANCY G. McPHEE       FEBRUARY 1980
30  '
40  CLEAR 1000: DEFINT I-T: DIM LAST(303,5): CLS
```

```
570 '
600 ' ********************************************************

610 'SUBROUTINE TO ENTER AND TOTAL POLL RESULTS
620 '
630 PRINT TAB(10) "(ENTER) POLL RESULTS"
640 INPUT"WHICH POLL"; I: IF I ) 303 THEN 640
650 INPUT "CRUDEN"; V1
660 INPUT "SHIPLEY"; V3
670 INPUT "WILSON"; V2
680 INPUT "OTHERS"; V4
690 VT = V1 + V2 + V3 + V4
700 POLL(I,1) = V1: POLL(I,2) = V2: POLL(I,3) = V3:
    POLL(I,4) = V4
710 POLL(I,5) = VT : NP = NP+1 : TV! = TV! + VT
720 IF NP = 1 THEN TV! = TV!-1
730 LT! = LT!+V1: PT! = PT!+V2: NT! = NT! + V3: OT! = OT!+V4
740 RETURN
750 '
800 ' ********************************************************
810 'SUBROUTINE TO DISPLAY INDIVIDUAL POLL RESULTS
820 '
830 PRINT TAB(10) "INDIVIDUAL POLL RESULTS"; TAB(45)
    "WHICH POLL";: INPUT I: IF I ) 303 THEN 830
840 PRINT "POLL # "; USING C$; I;: PRINT TAB(14) "CRUDEN";
    TAB(24); USING C$; POLL(I,1);
850 PRINT TAB(38) "1979:"; TAB(45) "LIB"; TAB(51); USING C$;
    LAST(I,1)
860 PRINT "----------"; TAB(14) "WILSON "; TAB(24); USING C$;
     POLL(I,2);
870 PRINT TAB(38) "-----"; TAB(45) "PC"; TAB(51); USING C$;
    LAST(I,2)
880 PRINT TAB(14) "SHIPLEY"; TAB(24); USING C$; POLL(I,3);
890 PRINT TAB(45) "NDP"; TAB(51); USING C$; LAST(I,3)
900 PRINT TAB(14) "OTHERS"; TAB(24); USING C$; POLL(I,4);
910 PRINT TAB(45) "OTH"; TAB(51) USING C$; LAST(I,4)
920 PRINT TAB(14) "TOTAL"; TAB(24); USING C$; POLL(I,5);
930 PRINT TAB(45) "TOT"; TAB(51); USING C$; LAST(I,5)
940 INPUT "(ENTER) TO CONTINUE"; X
950 RETURN
960 '
1000 ' *******************************************************
1010 ' SUBROUTINE TO CHANGE POLL FIGURES
1020 '
1030 INPUT "DO YOU WISH TO CHANGE POLL FIGURES (1/0)"; X
1040 IF X=0 THEN 1150
1050 INPUT "WHICH POLL"; I: IF I ) 303 THEN 1050
1060 IF I ) 303 THEN 1050
1070 IF F=2 THEN GOSUB 1170
1080 INPUT "LIBERAL"; V1
1090 INPUT "PC"; V2
1100 INPUT "NDP"; V3
1110 INPUT "OTHERS"; V4
1120 VT = V1 + V2 + V3 + V4
1130 IF F=1 THEN GOSUB 1200: GOTO 1030
1140 IF F=2 THEN GOSUB 1230: GOTO 1030
1150 RETURN
```

```
50  PRINT "DO YOU WISH TO:
60  PRINT "    1) ENTER POLL DATA          2) REVIEW A POLL"
70  PRINT "    3) RECORD DATA ON TAPE      4) END"
80  PRINT: INPUT "(ENTER) SELECTION"; X
90  CLS: ON X GOSUB 200, 400, 600, 810
100 CLS: GOTO 50
110 '
200 '*******************************************************
210 ' SUBROUTINE TO ENTER POLL DATA
220 '
230 PRINT "TYPE IN POLL FIGURES:"
240 PRINT "IF YOU MAKE A MISTAKE, RE-ENTER THE WHOLE POLL";
    : PRINT
250 INPUT "POLL # (999 TO END):"; I: IF I = 999 THEN 320
260 INPUT "LIB"; L : LAST(I,1) = L
270 INPUT "PC"; P: LAST(I,2) = P
280 INPUT "NDP"; N: LAST (I,3) = N
290 INPUT "TOTAL"; T : LAST(I,5) = T
300 O = T-(L+P+N) : LAST(I,4) = O : PRINT "OTHERS "; O
310 PRINT: GOTO 250
320 RETURN
330 '
400 '*******************************************************
410 ' SUBROUTINE TO REVIEW POLL DATA ON SCREEN
420 '
430 INPUT "WHICH POLL # DO YOU WISH TO REVIEW  (999 TO END)"; I:
    IF I = 999 THEN 460
440 PRINT I; TAB(10) LAST(I,1); TAB(20) LAST(I,2); TAB(30)
    LAST(I,3); TAB(40) LAST(I,4); TAB(50) LAST(I,5)
450 PRINT: GOTO 430
460 RETURN
470 '
600 '*******************************************************
610 ' SUBROUTINE TO ENCODE POLL DATA ON TAPE
620 '
630 PRINT "READY TO RECORD RANGE OF POLL FIGURES": PRINT
640 INPUT "WHAT IS LOWER POLL #"; R1: IF R1 ) 303 THEN 640
650 INPUT "WHAT IS UPPER POLL #"; R2: IF (R2 ( R1) OR
     (R2 ) 303) THEN 650
660 INPUT "FILE NAME"; F$
670 INPUT "(ENTER) WHEN READY TO TAPE (PRESS RECORD KEY)"; X
680 PRINT #-1, F$, R1, R2
690 R$ = "": T$ = "/"
700 FOR I = R1 TO R2
710     FOR J = 1 TO 5
720          IF LEN(R$ + STR$(LAST(I,J)) + T$) ) 230 THEN
             PRINT #-1, R$: R$ = ""
730          R$ = R$ + STR$(LAST(I,J)) + T$
740     NEXT J: PRINT I;
750 NEXT I: PRINT #-1, R$
760 PRINT: PRINT "DATA RECORDED FROM POLL"; R1; " TO"; R2
770 PRINT: INPUT "DOU YOU WANT TO RECORD MORE DATA (1/0)"; X
780 IF X=1 THEN 640
790 RETURN
800 '
810 CLEAR: CLEAR 50: END
```

Coding Data on Tape: Polldata and Lines 1300-1500

The evolution of these routines is a grisly example of learning from melancholy experience. When I wrote Election Night I had access to the poll results for the previous spring's election. Nothing easier, I thought, than to record that data on tape and play it back in on election night. I'd show them what this computer could do!

So I wrote POLLDATA, a little program to input the data and store it on tape. The input section, lines 200-320, was much like that in lines 600-750 of the main program, with one minor difference. My list gave the totals, but not the figures for OTHERS, so line 300 had to produce that information for me. Lines 400 to 470 permitted me to check the accuracy of the figures, which could if necessary be changed by the first routine.

Putting the data on tape seemed to present no problem. My original version was:

```
630 FOR I = 1 TO 303: FOR J = 1 TO 5
640 PRINT #-1, LAST(I,J)
650 NEXT J: NEXT I
```

To input the taped data into the program, I had merely to replace PRINT #-1 with INPUT #-1. What could be simpler? (And in fact, for disk storage, PRINT #1 and INPUT #1 inside nested loops would be a reasonably efficient method to store and retrieve the data, but use a semi-colon after LAST(I,J) to minimize disk space.)

Fortunately, before I had typed all the figures in, a random thought struck — would all that stuff fit on one 30-minute cassette tape? Using just the Play button on the recorder, I tried a dry run. Two and a half hours later, after much frantic turning over of tapes, the counter on the CTR-80 had passed 999 for the second time. Clearly some changes had to be made. So I added two new input variables, R1 and R2, to act as the lower and upper bounds of a range of polls. Line 630 was changed to read:

```
630 FOR I = R1 TO R2: FOR J = 1 TO 5
```

and I added PRINT I;: after NEXT J: in line 650, just to keep visual track of the recorder's stately progress. Some trial and error revealed that data for some 35 polls would fit on one side of a C-30 tape. Of course all the figures had to be retyped, and then it required 4 1/2 cassettes and nearly 3 hours of recording time to store all the data. Needless to say, I didn't make backups.

On the "great night" I arrived at headquarters some three hours early to set up and feed in my procession of tapes. By some abnormal stroke of luck, nothing ghastly occurred to foul things up, and the last tape finished just as the polls closed. But if there had been a tape misread, or the program had crashed, my goose would have been cooked — there would have been no possible way of reentering the tapes. I began to think longingly about disk drives.

Some weeks later (alas, too late) I ran across a little item in an old Radio Shack newsletter (July '79 in the US, November '79 in Canada) giving a technique for more efficient storage of tape data. The trick was to code and tape the data in long strings and then decode it on input. The Radio Shack program used a single-dimension array, but after some head-scratching I was able to adapt it to my two-dimension array.

What a difference! In the original version, recording 35 polls required 220-225 revs on the CTR-80 counter; the coded version took 12. The original took 14 minutes to record or load each group — the new version 45 seconds! A range of 70 polls loaded in 1 minute 20 seconds (compared to 12 seconds from a disk drive — disks are still the winners). I could have stored all my data on one side of a C-30, and loaded it in just over 5 minutes. We live and learn.

If you are going to store a lot of numerical data on tape, this is clearly one way to go. I suspect that some bright sparks out there may have devised even more efficient methods. If so, let's hope they'll share them with the rest of us.

My adaptation of the Radio Shack program is in lines 600-760 of POLLDATA (to code and record the data) and line 1300-1500 of Election Night (to input and decode).

Line 680 prints on tape an identifying filename and the lower and upper poll numbers of the range being recorded. Line 690 sets up 2 strings; R$, into which the data will be gathered, is initialized as a null string; T$ is a slash, which will be used to separate the data elements. Lines 700-710 set up the same nested loops as before.

Now the fun begins. Look at line 730 first to see what happens to each data item. The current data item (LAST(I,J)) is converted into a string, and added to whatever is already in R$, followed by a slash. Before each new piece of data is added, however, line 720 checks to see that the new addition will not make R$ longer than 230 characters. If it will, the current contents of R$ are printed on tape, R$ is set back to null and the process of adding data items starts over again until all the data has been recorded or added to the current R$. Line 750 takes care of any leftover data in R$ after the last poll has been added.

Notice that line 40 of POLLDATA clears 1000 bytes of string space. By all logic only 300 should be needed to store R$ and the filename, but I kept getting "out of string space" errors until I cleared at least 700-750. There was no such problem in Election Night.

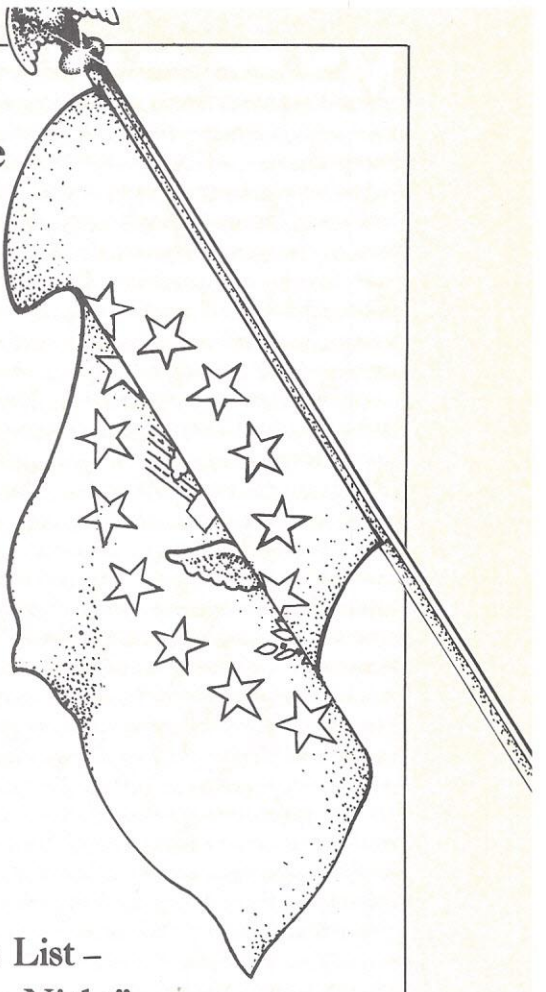
The decoding routine in lines 1300-1500 is a little more complex. Lines 1330-1360 input the filename and the lower and upper poll numbers from tape, and allow the option to change the tape if an error has been made. Line 1380 sets I at the lower poll number, sets J at the first piece of data and initializes R$ and T$ as before. Line 1390 inputs a data string from tape and initializes a new variable, SP, which points to the starting position in the string of the current data item.

Lines 1400-1450 set up a loop which looks in turn at each character in R$. In line 1410, if the current character (MID$(R$,K,1)) is not a slash the routine passes to NEXT K and looks at the next character; if it is a slash (indicating the delimiter between data items) then the number of characters (NC) in the current data item is equal to the current value of K minus the position of the SP pointer. For example, for the first piece of data, SP is at 1, the beginning of the string. If the slash is found at the fourth character, the number of characters in the data item is equal to 3.

Line 1420 converts (VAL) the string characters containing the current data item to numeric value and stores this in the LAST array. Line 1430 moves the SP pointer to the next character after the slash and increments J by 1 (to the next piece of data for this poll); if J is now greater than 5 (the last data item in the poll) it sets J back to 1 and increments I to the next poll number. Finally it prints the new poll number on the screen to allow you to keep track of the program's progress. Line 1450 completes the loop, parsing the entire string in the same fashion; then R$ is set to null again and the next data string is brought in. Line 1440 checks to see if the poll number is now higher than R2, indicating the end of the data and if so goes to 1460. Successive ranges of polls can be input until the data is complete. Even though much more data can be recorded by this method, it is probably still sensible to record and input it in sections, and of course, making-backups of your data tapes is now a snap. □

```
            ETOBICOKE CENTRE -- ELECTION RESULTS
    NUMBER OF POLLS REPORTING:  96 / 303  ( 31.7 % OF TOTAL)
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              CRUDEN  (LIB)        8,523    ( 42.2 %)
              WILSON  (PC)         8,274    ( 41.0 %)
              SHIPLEY (NDP)        3,129    ( 15.5 %)
              OTHERS                 251    (  1.2 %)
              TOTAL VOTE:         20,177
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
       SAMPLE RUN:  #1 -- Current totals on top half of screen;
                          waiting for INKEY$ input.
            ETOBICOKE CENTRE -- ELECTION RESULTS
    NUMBER OF POLLS REPORTING:  96 / 303  ( 31.7 % OF TOTAL)
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              CRUDEN  (LIB)        8,523    ( 42.2 %)
              WILSON  (PC)         8,274    ( 41.0 %)
              SHIPLEY (NDP)        3,129    ( 15.5 %)
              OTHERS                 251    (  1.2 %)
              TOTAL VOTE:         20,177
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
         (ENTER) POLL RESULTS
WHICH POLL? 58
CRUDEN? 123
SHIPLEY? 24
WILSON? 113
OTHERS? 2
      SAMPLE RUN:  #2 -- Entering new poll results
            ETOBICOKE CENTRE -- ELECTION RESULTS
    NUMBER OF POLLS REPORTING:  97 / 303  ( 32.0 % OF TOTAL)
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              CRUDEN  (LIB)        8,646    ( 42.3 %)
              WILSON  (PC)         8,387    ( 41.0 %)
              SHIPLEY (NDP)        3,153    ( 15.4 %)
              OTHERS                 253    (  1.2 %)
              TOTAL VOTE:         20,439
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
          INDIVIDUAL POLL RESULTS              WHICH POLL? 58
POLL #  58    CRUDEN    123          1979:  LIB   107
----------    WILSON    113          -----  PC    120
              SHIPLEY    24                 NDP    29
              OTHERS      2                 OTH     8
              TOTAL     262                 TOT   264
(ENTER) TO CONTINUE? _
      SAMPLE RUN:  #3 -- Totals updated; display individual
                         poll and compare results with
                         previous election.
```

Sample Run



Variable List – "Election Night"

A$, B$, C$: These are formatting strings as described elsewhere.

LT!, NT!, OT!, PT!: Total votes for Liberal, NDP, Others and Progressive Conservative, respectively.

TV!: Overall total vote

NP: Number of polls reporting

POLL: Array which stores the poll results as they are input.

LAST: Array which stores former results input from tape.

V1, V2, V3, V4: Individual poll figures for each candidate. These are temporary variables.

VT: Total vote in a poll. Another temporary variable.

I, J, K: Index variables used in loops and as pointers. I invariably points to the poll number; J to the data items within each poll.

F: Used as a flag

X, X$: Input response variable; used when some response is required to continue. X$ is used in the INKEY$ routine.

Used Only In Tape Routine:

F$, F1$: Filenames

R1, R2: Lower and upper range of polls to be input.

R$: String containing coded data.

T$: Delimiter between items of data within R$.

SP: Starting position within R$ of current data item.

NC: Number of characters in current data item.

Computer Cryptography for Coded Communications

BY JOHN H. HEIDEMA

Many people often require communications security not available in a phone conversation, a telegram or a written message. This need is obvious in the conduct of war and diplomacy. It is equally obvious for business and banking operations, though it may not always be readily and cheaply available.

Here's a fairly simple but secure encoding and decoding process (encryption and decryption) using a microcomputer. The program, written for a TRS-80 with Level II Basic, should run with minor modifications on most micros.

Encoded messages have been used since ancient times, but the ingenuity of codebreakers (cryptanalysts) has generally kept pace with developments in coding techniques. Unlike the simple monoalphabetic substitutions used in ancient coded messages and in the cryptograms found in modern puzzle magazines (see "Solving Ciphers," June *PC*), today's best codes are polyalphabetic. That is, each successive letter is encoded with a different substitute alphabet. The various substitute alphabets being used and the order of their use is presumably known only to the encoding and decoding parties. Surprisingly even this encryption technique succumbs to diligent cryptanalysis if the volume of messages is high and the sequence of substitute alphabets used is repeated. (Computerized methods greatly aid and speed such cryptanalysis). But using many substitute alphabets in a random, never repeating sequence known only to to the encoder and decoder leads to totally secure encoded communication with no "handle" for the cryptanalyst to grasp. An ordered series of random numbers known to sender and receiver can serve to select the order of alphabets used.

The encryption and decryption program I describe here generates 37 different substitute alphabets. It also permits sender and receiver to use their computers to select a series of random numbers. Using the same model of computer, they can both have access to the same identical, ordered set of random numbers (from an almost innumerable number of such sets available); and they should then be able to communicate rapidly and securely.

The program in operation is pretty much self-instructive. It has four basic parts, or segments: 1) the beginning, including an alphabet generating routine; 2) the encryption section; 3) the decryption section; and 4) the subroutine to create and store the list of random numbers.

The program uses the following char-

Program Listing

```
10 REM   SUPERDUPER ENCRYPTION AND DECRYPTION PROGRAM
                  BY JOHN H. HEIDEMA  (JAN./FEB., 1980)
20 CLS:PRINT:PRINT:PRINT:CLEAR 1500
30 INPUT"MAXIMUM NO. OF CHARACTERS (OR RANDOM NUMBERS)";S
40 DIM A$(40), C$(S), N(S)
50 A$(1)=";":FOR I=2 TO 11:A$(I)=CHR$(I+46):NEXTI
60 FOR I=12 TO 37:A$(I)=CHR$(I+53):NEXTI
90 CLS:PRINT:PRINT
100 INPUT"ENCRYPTION OR DECRYPTION ";A$
110 IF LEFT$(A$,1)="D" THEN 1000
150 GOSUB 2000
190 PRINT:PRINT
200 PRINT"   ENTER EACH CHARACTER ONE AT A TIME IN"
210 PRINT"SEQUENCE.  ENTER 'END' WHEN FINISHED.  YOUR"
220 PRINT"CODED MESSAGE WILL THEN BE PRINTED OUT."
230 PRINT:PRINT
250 C=0:P5=0
300 INPUT"REAL CHAR. ";C$
310 IF C$="END" THEN 700
320 IF LEN(C$)<>1 THEN PRINT"INCORRECT ENTRY.";
    "  REENTER LAST CHARACTER.":GOTO 300
330 C=C+1
340 R=N(C)-INT(N(C)/37)*37:IF R=0 THEN R=37
350 FOR I=1 TO 37
360 IF A$(I)=C$ THEN E=I:I=38:GOTO 400
370 NEXTI
380 PRINT"NOTE:  A WRONG CHARACTER HAS BEEN ENTERED.
400 F=E+R:IF F>37 THEN F=F-37
410 C$(C)=A$(F):C$=""
450 GOTO 300
700 REM      PRINTOUT OF CODED MESSAGE
720 PRINT:PRINT:PRINT"CODED MESSAGE:":PRINT
740 FOR I=1 TO C:LPRINT C$(I);
750 IF I/5 = INT(I/5) THEN LPRINT " ";:P5=P5+1
760 IF P5=6 THEN LPRINT " ":P5=0
780 NEXTI
900 GOTO 9999
1000 REM   DECRYPTION SECTION
1100 GOSUB 2000
1190 PRINT:PRINT
1200 PRINT"   ENTER EACH FALSE CHARACTER ONE AT A TIME "
1210 PRINT"IN SEQUENCE.  ENTER 'END' WHEN YOU ARE FINISHED.
1220 PRINT"YOUR DECODED MESSAGE WILL THEN BE PRINTED OUT."
1230 PRINT:PRINT
1250 C=0
1300 INPUT"FALSE CHAR. ";F$
1310 IF F$="END" THEN 1700
1320 IF LEN(F$)<>1 THEN PRINT"INCORRECT ENTRY.";
     "  REENTER LAST CHARACTER.":GOTO 1300
1330 C=C+1
1340 R=N(C)-INT(N(C)/37)*37:IF R=0 THEN R=37
1350 FOR I=1 TO 37
1360 IF A$(I)=F$ THEN F=I:I=38:GOTO 1400
1370 NEXTI
1380 PRINT"NOTE:  WRONG CHARACTER HAS BEEN ENTERED."
1400 E=F-R:IF E<1 THEN E=E+37
1410 C$(C)=A$(E):F$=""
1450 GOTO 1300
1700 REM      PRINTOUT OF DECODED MESSAGE
```

```
1720 PRINT:PRINT:PRINT"DECODED MESSAGE:":PRINT
1740 FOR I=1 TO C:IF C$(I)=";" THEN LPRINT" ";:GOTO 1760
1750 LPRINT C$(I);
1760 IF I/36 = INT(I/36) THEN LPRINT" "
1780 NEXTI
1900 GOTO 9999
2000 REM   SUB TO GENERATE AND STORE RANDOM NUMBERS
2010 PRINT:INPUT"WILL YOU PROVIDE THE RANDOM NUMBERS ";A$
2020 FOR I=1 TO S:N(I)=0:NEXTI
2030 IF LEFT$(A$,1)="Y" THEN 2200
2100 RS=1    :REM  USE RS=1 FOR MACHINE RND() METHOD
                   USE RS=2 FOR SINE METHOD
                   USE RS=3 FOR LOG METHOD
                   USE RS=4 FOR ALGEBRAIC METHOD
                   USE RS=5 FOR SQUARE ROOT METHOD
2140 PRINT
2150 ON RS GOSUB 2300,2500,2700,2900,3100
2190 GOTO 2290
2200 PRINT:PRINT
2210 PRINT"OK.  ENTER THE PROPER SEQUENCE OF RANDOM"
2220 PRINT"NUMBERS AS THEY ARE REQUESTED. "
2230 PRINT"ENTER  0 (ZERO) TO TERMINATE ENTRIES."
2240 PRINT:FOR I=1 TO S
2250 PRINT"  RANDOM #";I;:INPUT R
2260 IF R=0 THEN I=S+1:GOTO 2290
2270 N(I)=R:NEXTI
2290 RETURN
2295 REM   ALTERNATIVE SUBROUTINES FOR
           RANDOM NUMBER GENERATING:
2296 REM
2300 REM  SUB USING MACHINE RND() FUNCTION
2305 S5=173
2310 FOR I=1 TO 3:POKE 16553+I,S5+7*I:NEXT I
2315 INPUT"MAXIMUM SIZE FOR RANDOM NUMBERS ";M
2320 INPUT"NUMBER OF LEADERS TO DISCARD ";L
2325 PRINT"RANDOM NUMBERS BEING GENERATED NOW."
2330 FOR I=1 TO L:R=RND(M):NEXT I
2340 FOR I=1 TO S:N(I)=RND(M):NEXT I
2350 RETURN
2500 REM  SUB FOR SINE METHOD
2510 INPUT"RANDOM NUMBER SEED ";S5
2520 INPUT"NO. OF LEADERS TO DISCARD ";L
2530 IF S5>10 THEN S5=S5/10:GOTO 2530
2550 PRINT"OK.  RANDOM NUMBERS BEING GENERATED NOW."
2600 S5=S5+L*.0019153
2610 FOR I=1 TO S:S5=S5+.0019153:F1=SIN(S5)+1
2620 F=10000*F1-INT(10000*F1)
2640 N(I)=INT(F*1000)+1:NEXT I
2650 RETURN
2700 REM  SUB FOR LOG METHOD
2710 INPUT"RANDOM NUMBER SEED ";S5
2715 PRINT"RANDOM NUMBERS NOW BEING GENERATED."
2720 S5=S5/2.718 - INT(S5/2.718) + 1.3
2730 FOR I=1 TO S:S5=S5+.01002211
2740 S6=LOG(S5)
2750 F=10000*S6-INT(10000*S6)
2760 N(I)=INT(1000*F)+1 :NEXT I
2770 RETURN
2900 REM  SUB FOR ALGEBRAIC METHOD
2910 DEFDBL Z
2920 Z1=1024*1024+3
2930 Z2=1024*1024*1024*1024
2935 REM          ALGORITHM REQUIRES Z2=2↑P
                  AND  Z1=2↑(P/2)+3  WHERE
                  P IS AN EVEN INTEGER.
2940 INPUT"RANDOM NUMBER SEED ";Z5
2945 PRINT"OK.  RANDOM NUMBERS NOW BEING GENERATED."
2950 Z5=10*Z5+7
2960 FOR I=1 TO 25:Z5=Z5*Z1
2970 Z5=Z5-INT(Z5/Z2)*Z2:NEXT I
2980 FOR I=1 TO S:Z5=Z5*Z1
2990 Z5=Z5-INT(Z5/Z2)*Z2:F=Z5/Z2
3000 N(I)=INT(F*1000)+1:NEXT I
3010 RETURN
3100 REM  SUB FOR SQUARE ROOT METHOD
3110 INPUT"RANDOM NUMBER SEED ";S5
3115 PRINT"OK.  RANDOM NUMBERS NOW BEING GENERATED."
3120 IF S5>50 THEN S5=S5/10:GOTO 3120
3130 FOR I=1 TO S:S5=S5+.001039163
3140 S6=SIN(S5):F=10000*S6-INT(10000*S6)
3150 N(I)=INT(1000*F)+1 : NEXT I
3160 RETURN
9999 END
```

acters (in this order) in its "alphabet":

;123456789ABCDEFGHIJKLMNOP QRSTUVWXYZ.

These total 37. The semicolon should be used during encryption to represent spaces between words and sentences. No other punctuation is incorporated. Probably the best method is one space between words and two spaces between sentences. The ordered series of random numbers determines which of these characters begins the substitute alphabet. Thus if 'Q' begins the substitute alphabet, then Q = ; and P = Z. An awesomely large number of other alphabet sizes and arrangements are possible. Switching to a randomly mixed sequence of these 37 characters might possibly increase security (e.g., ...-M-6-3-W-B-9-...).

When encryption (lines 150 to 900) begins, you enter characters one at a time. Again, use a semicolon for spaces. Enter "END" to terminate the entry phase. Your encrypted message will then be printed out in normal code format. All LPRINT statements in this program should be changed to PRINT for other computers or if no printer is attached to your TRS-80.

If you select the decryption mode (lines 1000 to 1900), the false characters are also entered one at a time (but no semicolon for spaces this time). Again enter "END" to terminate entries. Your properly spaced message will then be printed out. Use PRINT if LPRINT isn't appropriate. You can change the number 36 in line 1760 to a larger number if your printer legibly prints more than 40 characters per line.

The random number generating subroutine (lines 2000+) produces a sufficiently large fixed sequence of random numbers in array N( ). You may enter your own sequence of numbers manually or you may use random numbers selected by the computer. (The encryption and decryption sections convert these to numbers from 1 to 37 to determine which alphabet to use.) The method for generating a reproducible series of random numbers by computer ("pseudorandom" numbers) is flexible. You can, of course, use the Basic RND function. That was my initial choice as I first wrote this program, and the program as presented here uses the RND function (lines 2300 to 2350). The algorithm employed gives, I think, appreciable security. I have checked other TRS-80 Level II machines and they all generate the *same* random numbers using this algorithm, even with disk Basic. Lines 2305 and 2310 reseed the Level II random number

Sample Run

```
MAXIMUM NO. OF CHARACTERS (OR RANDOM NUMBERS)? 20

ENCRYPTION OR DECRYPTION ? E

WILL YOU PROVIDE THE RANDOM NUMBERS ? NO

MAXIMUM SIZE FOR RANDOM NUMBERS ? 631
NUMBER OF LEADERS TO DISCARD ? 317
 RANDOM NUMBERS BEING GENERATED NOW.

   ENTER EACH CHARACTER ONE AT A TIME IN
SEQUENCE.  ENTER 'END' WHEN FINISHED.  YOUR
CODED MESSAGE WILL THEN BE PRINTED OUT.
REAL CHAR. ? S
REAL CHAR. ? E
REAL CHAR. ? L
REAL CHAR. ? L
REAL CHAR. ? ;
REAL CHAR. ? I
REAL CHAR. ? B
REAL CHAR. ? M
REAL CHAR. ? END
CODED MESSAGE:  U5EA9 TDJ
```

```
MAXIMUM NO. OF CHARACTERS (OR RANDOM NUMBERS)? 30

ENCRYPTION OR DECRYPTION ? D

WILL YOU PROVIDE THE RANDOM NUMBERS ? NO

MAXIMUM SIZE FOR RANDOM NUMBERS ? 631
NUMBER OF LEADERS TO DISCARD ? 317
 RANDOM NUMBERS BEING GENERATED NOW.

   ENTER EACH FALSE CHARACTER ONE AT A TIME
IN SEQUENCE.  ENTER 'END' WHEN YOU ARE FINISHED.
YOUR DECODED MESSAGE WILL THEN BE PRINTED OUT.
FALSE CHAR. ? U
FALSE CHAR. ? 5
FALSE CHAR. ? E
FALSE CHAR. ? A
FALSE CHAR. ? 9
FALSE CHAR. ? T
FALSE CHAR. ? D
FALSE CHAR. ? J
FALSE CHAR. ? END
DECODED MESSAGE:  SELL IBM
```

generator so that the machine will give the same sequence of random numbers on each identical run of the program. Without these two lines the computer would have to be turned off and the program reloaded to reproduce a given random number sequence. Line 2305 and 2310 should, of course, be modified or deleted if the program is used on a different computer.

Despite the available RND function and a decent algorithm to exploit its power, I was concerned for security in the face of determined cryptanalysis (the N.S.A. can solve almost anything!). I therefore developed four other algorithms which use the computer to rapidly generate random numbers, or numbers sufficiently random for these purposes. Each is presented as a subroutine at the end of the program. You need only change line 2100 (or add a line, e.g. 2145 RS=3) to access them. Three of these four subroutines utilize some of the different types of irrational numbers computable with Level II Basic; sines, logarithms and square roots are employed. I have found that such irrational numbers are sufficiently "normal" (in the mathematical sense) that a selection of later digits is essentially random. One subroutine computes reproducible random numbers by an algebraic algorithm.

I should perhaps point out that the RND algorithm generates random numbers several times as fast as the four alternatives here, but random numbers at 300 to 500 per minute shouldn't slow anybody down much.

Individual users could select any (or all?) of these alternative subroutines as is or with their own modifications (e.g., S5 in line 2305 may vary from -7 to +234; and the rather strange constant values in lines 2600, 2610, 2730 and 3130 are arbitrary and up to the user.) Modifications like these should give users such a variety of random number possibilities that cryptanalysis (without other information besides this article) must fail.

Communicating users will have to agree beforehand on either which "seeds" to use for the random number generators or where and how to incorporate this "key" information into the cryptogram. For example, put the maximum size in the second group from the front and the number of leaders to discard in the second group from the rear; and let letters equal their alphabet sequence number equivalent (mod 10; e.g. AZ3BD=16324). Thus, using this particular method for coding in the "key" numbers the message "Will 800,000 bushels be satisfactory?" becomes:

L12XM A2GB9 YLL9V W9WB4
DQT3F 4GOPF 0CYU; J0CB4 E3JQG

The underlines, indicating the key information, should *not* normally be included. Try it. It should work just like this on a TRS-80.

Here are some other messages you can try. They should suggest applications and methods, and they should also help you to debug your keyed in program: "Strong competition. Reduce bid to 215,000" becomes:

;XXX5 RAP5F V3OLA G9H7P 3JEY4
0L;G8 WX6;S K6PJJ

[if you use line 2145 RS=2 (sine method) and the key numbers 22943 and 46151] Also using line 2145 RS=4 (the algebraic method) with the seed number 9,563,838,651 you can decipher this message:

UQY05 KZXXP 5SOH; 4VX9A C9;A9 L7WMM 6IPCP XND6K YLYLL 0W2T4 E0ZHH 36ZOH X2I2Q B7X;L A194D OZ6JW XWJR2 838T8 VY452 PDLU8 CEY3Y MQU;Q KXL4N JVBCA NY0ZD VOYLJ

as: TRANSFER FROM ACCT NO E1640 REPEAT E1640 TO ACCT NO E3942 REPEAT E3942 THE SUM OF 95000SF REPEAT 95000SF. AUTHORIZATION: H. P. SNEED

Modifying and segmenting this program will permit it to function easily on 4K machines. My original programs were less than 1K each, doing encryption and decryption in separate programs with the random numbers and corresponding characters entered as pairs. The complete listing occupies about 4K, although removing all documentation and unused subroutines would reduce that significantly.

A program like this one is, of course, also usable on larger computers. (I have it running on a DEC PDP11/34A with very few modifications.) It seems to me that the privacy of disk file information (on big systems or small) could be insured by having the data run in and out through a program similar to this one.

This program can no doubt be modified to accept and deliver data through a modem connection. With the use of string functions it could be improved to accept whole lines of text for line-by-line encryption and decryption. Such modifications could lead to very fast, secure communications.

After I had a version of this program running, I obtained and read a copy of *The Codebreakers* by David Kahn (Macmillan, 1967). Though not new any more, the book is a very well written and well documented historical survey of cryptology. It gave me considerable pleasure and some insights into cryptology which helped me to improve the quality of this program. I highly recommend the book to you. □

How to write for Personal Computing

You've written the programs we want to publish. You – the *Personal Computing* readers – are using your computers in businesses, homes, offices and schools. Other readers, just as software-hungry as you, are eager to try out your programs, your applications and your techniques. So why not share what you've done by submitting an article to *PC?*

It's easier than you might think. Remember: we're more interested in practical programs and useful applications than in fancy prose. And our editorial staff stands ready to help with any problems you encounter in writing your article; just give us a call at (617) 232-5470.

Here are some handy guidelines to help you get started.

First, decide what kind of article you want to write. Do you have a *business program* that will help an executive, salesman, doctor, lawyer or shopkeeper function more efficiently? Think about how businesses can benefit from microcomputers – not only in the obvious areas of inventory, accounting and payroll, but in all departments and levels right up to the president's desk. Financial and marketing analysis, time management, planning, material handling, product design and cost accounting are areas ripe for creative programming.

How do you use your computer for *home and personal applications* in your living room, kitchen, study or den? Again, think beyond the obvious areas of checkbook balancing and budgeting (though these areas are far from exhausted) to other applications. Hobbies, home management, household inventory, gardening and landscaping, personal income and expense analysis, personal mailing lists and word processing are just a few ideas to spark your imagination.

What *education programs* have you written for children, adults, professionals, businessmen and teachers? Computers can not only teach children basic subjects such as spelling, math, geography, economics, civics, grammar, literature and science, but can help adults review or sharpen skills in these areas as well. How else can computers function in or out of the classroom to aid learning? To help teachers and administrators?

Are you proficient in some programming technique or special computer area you could explain in a *tutorial article*? How do you save time, money, computer memory or frustration when programming or using your computer? Others can benefit from the same techniques you use.

Computer games, history, humor and *fiction* are other areas rich in article and story ideas.

Your second step is to write the text of the article. Remember, readers aren't familiar with your program. So explain in detail what the program does and how it does it. Include here the overall structure of your program as well as any special algorithms or routines you've used. Give suggestions for modifying or expanding the program for other applications, other businesses or other situations.

Third, prepare your supporting documentation. Include at least a program listing and one or two sample runs, and add program notes to explain any special commands used or other special features of your program. Use charts, diagrams, figures and photos if they help explain your program and its use.

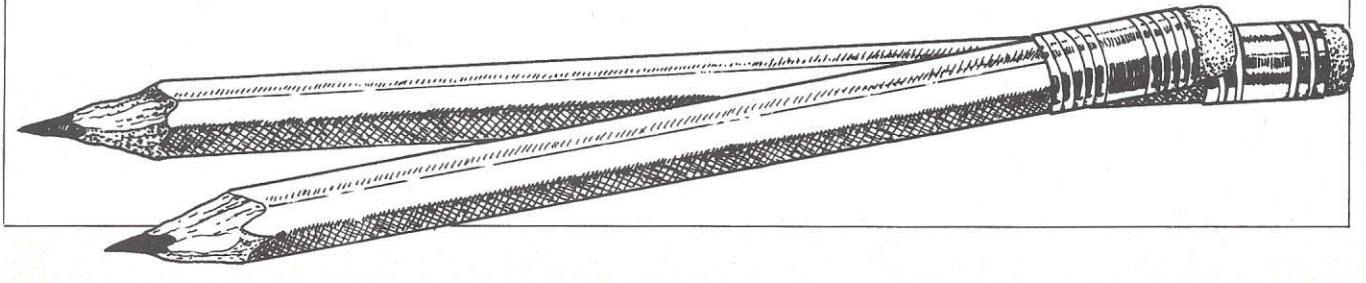
Finally, mail your manuscript. Address it to: Don Wood, Managing Editor, Personal Computing Magazine, 1050 Commonwealth Ave., Boston, MA 02215.

A few suggestions: All submissions should be original, typed (*not* all CAPS), double-spaced and neat. Please include your name and address on the first page of the article and enclose a self-addressed, stamped envelope for return of material.

Since we photograph program listings and sample runs exactly as you send them to us for publication in the magazine, please be sure you use a fresh ribbon for computer printouts. If you don't have a printer, you can type your listings single spaced; but again, be sure you use a new ribbon. (If your program relies heavily on graphics, you can photograph sample runs from your CRT. But take care to avoid distortion due to the curve of the screen.)

Feel free to call us if you have any questions or want to discuss specific ideas. We can give you feedback and suggest appropriate slants and approaches.

We're always looking for fresh, original ideas. While these guidelines will help you in preparing material for *Personal Computing*, don't assume we don't want your idea just because it's not mentioned here. Let us and our readers know what *you're* doing with your computer.



Printing and Storing Weekly Schedules

BY IVAN FLORES

My lifestyle requires the things I do to change from one week to the next. It is possible to keep track of these changes manually, but I have a North Star computer, so why not let it do the work? This program produces a weekly schedule on one page.

As each new appointment comes up, I want to enter it quickly and, of course, delete appointments which no longer apply. I want to print a schedule in a few seconds and store it on a disk so I can later call it up for display and alteration at a moment's notice.

My first problem was to decide how much to display and what to do about a.m. and p.m. hours. I decided that a 12 hour calendar would be sufficient and would do away with a 24 hour display; it would also simplify the program.

When I first constructed the program, I had only one file for one schedule but it only recorded data for one week. It did not supply the convenience to plan ahead. The latest program provides four files called A, B, C and P.

The P file is a prototype. It is a copy of what my week would generally be. This way, I do not have to start from scratch each time I prepare a new schedule. I take the prototype and alter it to create a new schedule. Of course, the prototype itself can be altered.

There are three active schedules called A, B and C. Suppose, for example, that schedule A is for the week of May 5th (I always start the week on Monday). Then schedule B is for the week of May 12th and schedule C for the week of May 19th.

As May wears into June, besides new flowers, we see a new schedule. The schedule for May 5th has expired and I now use schedule A for the week of May 26th.

There are four files, one for the prototype and one for each of the schedules. Each file has just three fields, labelled S$, M$ and E, in that order:

- S$ is a string which contains the entire schedule.
- M$ is the month to which the schedule applies.
- E is the entry date for the Monday to which the schedule applies.

A sample printout is presented in the Sample Run. At the top is a header announcing this as a schedule and tells the week and day to which it applies. There are eight column heads, one for each day of the week and, at the extreme left, a time indicator. The schedule is constructed in quarter-hour intervals which start at 9 a.m. and continue to 8:45 p.m. An appointment appears as a multiple of fifteen-minute intervals in a column for the appropriate day.

Each entry is a string of up to ten characters. Nine characters are a preferable maximum or there will not be a space between one entry and the next. The schedule displays 12 hours in quarter-hour intervals. Hence, there are 48 rows, each with eight entries giving you a total of 384 entries. At 10 characters per entry, you will have 3840 characters. All this information is held in a single string called S$. Strings of more than 10 characters must be declared in my North Star Basic and this is done in (50). (The numbers in parentheses refer to line numbers in the Program Listing.)

Professor Flores teaches statistics at the Baruch College, City University of New York where he has been teaching Computer Methodology since 1967.

Displays and Printouts

The program produces schedules in two forms — display and printout. I carry around the printout to remind me of commitments and my remaining free time in the next three weeks. I can also make notes on it for future entry to produce a new schedule. But while I am modifying an existing schedule at the console, I need a display of what that schedule currently contains.

Let us now see how a schedule display is produced (94-180). Assume that the string S$, contains an updated schedule which is to be displayed. It cannot be presented on the CRT as one unit because it is 48 lines long. It could be shown in two parts of 24 lines each, but there would be no room for column labels at the top. Hence, I choose to display it in three parts. The heading appears for each of the three successive displays. To go from one to the next, the carriage return is pressed on request (180).

First, the line number index (J) is set to 0 (94). A header is then displayed presenting the date of the schedule and a column heading for the time and days of the week (96-110). A loop is now entered to display 16 lines at a time each with 80 characters extracted from within the string (120-140).

After one full screen is presented, J is incremented by 16 pointing to the next set of 16 lines (160). Check to see if the whole schedule is presented (170). If not, you can hit the carriage return for more (180).

The North Star substring notation allows us to extract a set of characters at any point in the string. It uses the notation S$(B,E) to request that a substring starting at B and ending at E is withdrawn. The "print" request (130) (! abbreviates PRINT) displays one properly formatted line of the schedule. The starting address uses the subscript I, multiplied by 80, so as the loop progresses, we advance to a new line. The first line starts with byte number 1, not 0, the second with 81 and so forth. The beginning address is 80 times I plus 1. The screen does not properly accommodate 80 characters and so only 77 are displayed.

Another routine prints out the schedule A (390-450) on the printer. At the top is the header containing the month and Monday's date (390,400), then dates for the rest of the days of the week (404,406). This works fine except for the dates at the end of the month. For instance, if Monday is the 30th, Tuesday will always be the 31st, Wednesday will be the 32nd and so forth.

Labels for the columns are printed next (410-420). A loop then produces 48 lines which comprise the schedule proper. This is similar to the display loop, except the printer has no problem handling 80 character lines and does the whole thing without operator intervention.

Sample Run

```
                        S   C   H   E   D   U   L   E

              A                   Week  of  April   21

              21         22         23         24         25         26       27

Time      Monday     Tuesday    Wednesday  Thursday   Friday     Saturday  Sunday
 9:00                 phil                                      7:45
                      phil                             Yoga      Phila
                      phil                             Yoga      Phila
                                                 Diane Yoga      Phila
10:00     ########   Rose                       Diane Yoga      Phila
          ########   Rose                       Diane Yoga      Phila
          ########   Rose                       Diane Yoga      Phila
          ########   Rose                       Diane Yoga      Phila
11:00     Chorny     Rose                       Diane Yoga      Phila
          Chorny     Rose                       Diane Yoga      Phila
          Chorny     Rose                       Diane           Phila
                     Rose                       Diane           Phila
12:00                Rose                       GnlFclty        Phila
                     Rose                       GnlFclty        Phila
                     Rose                       GnlFclty        Phila
                     Rose                       GnlFclty        Phila
 1:00                Rose                       GnlFclty        Phila
                     Rose                       GnlFclty        Phila
          WEBER      Rose                                       Phila
          WEBER      Rose                                       Phila
 2:00     WEBER      Rose                                       Phila
                     Rose                                       Phila
 2:30                Rose                                       Phila
                     Rose                                       Phila
 3:00                                                           Phila
          Fishman                                               Phila
          Fishman                            dominic            Phila
          Fishman                            dominic            Phila
 4:00     =========            Fishman       dominic            Phila
          =========            Fishman       phil               Phila
          =========            Fishman       phil               Phila
          =========            #########     phil               Phila
 5:00     =========            #########     elliot             Phila
          =========            #########     elliot             Phila
 5:30     Op Systms ThesisAlt  Assmblr       elliot             Phila
          Op Systms ThesisAlt  Assmblr       =========          Phila
 6:00     Op Systms ThesisAlt  Assmblr       doreen             Phila
          Op Systms ThesisAlt  Assmblr       doreen             Phila
          Op Systms ThesisAlt  Assmblr       doreen             Phila
          Op Systms ThesisAlt  Assmblr                          Phila
 7:00     Op Systms ThesisAlt  Assmblr                          Phila
          Op Systms ThesisAlt  Assmblr
          Op Systms ThesisAlt  Assmblr
          Op Systms ThesisAlt  Assmblr
 8:00                                                           Mensa
 8:15                                                           Mensa
```

Entry Routine

A most important function of the program is accepting schedule changes with facility. The first version of the entry requires that you separately enter the four parameters required for each change. Now they are all entered on the same line (210-280). You enter the date of the new appointment, its time, the number of quarter-hour intervals that it occupies and a string of up to 10 characters to describe the appointment. These have the variable names, respectively, D, T, Q and A$.

The problem is to find the proper place in S$ to enter the appointment information. The time, T, is in hours and minutes. (Of course, we expect minutes to be 00, 15, 30 or 45). These units are inconvenient for the program to deal with so it is necessary to convert hours and minutes into minutes alone. For instance, an appointment time of 10:15 a.m. has a value in minutes of 615. The conversion is done by taking the number of hours, multiplying by 60 and adding the number of minutes (290). Another way to look at this is to consider the time in hours and minutes as an integer and to subtract 40 times the number of hours from it. This gives the proper result after ELSE (290).

Now we have another problem. The times we are interested in range from 9 a.m. to almost 9 p.m. A time between 9 a.m. and 12:45 p.m. is converted as described above. A time between 1 p.m. and 8:45 p.m. is off by 12 hours. But 12 hours is 720 minutes so for times registering less than 900, the conversion formula adds 720 (at the left of 290).

Now that we have recalculated T as time in minutes, it is necessary to convert this into a line number. Observe that 540 minutes is equivalent to 9 o'clock and will be on line 0. Each

15 minute interval will add one to the line number, L (300).

Now we have located the line according to appointment time. Next, move along the line, according to the day, in increments of 10 characters. The beginning byte address in the string of the insertion address, B, is determined as follows: multiply the line number, L, by 80, add 10 times the number of the day of the week, D, and add 1 to put us at the proper beginning of the line.

The method above (300) leads us to a byte address corresponding to the appointment time. The appointment data,

Program Listing

```
10REM********** SCHED rev 6/3/80 ******************
20REM SCHED IS A PROGRAM TO CONSTRUCT WEEKLY SCHEDULES
30!"  ENTER SYMBOL FOR OLD SCHEDULE FILE"
32!"     P FOR PROTOTYPE OR A, B OR C:  "\F$=INCHAR$(0)\!F$
34!"     SIMILARLY ENTER SYMBOL FOR OUTPUT SCHEDULE FILE:  "\T$=INCHAR$(0)\!T$
38OPEN#1, "SCHD" + F$ + ",2"
50DIM S$(4000), D$(20)
60READ #1,S$,M$,E
62!"     This schedule is for ",M$,"   ",E
64!"DO YOU WANT TO CHANGE THIS DATE?  "\Y$=INCHAR$(0)
66IF Y$ <> "Y" THEN GOTO 80
68INPUT "ENTER NAME OF THE MONTH:  ",M1$
70INPUT "ENTER MONDAY'S DATE:  ",E1
72E=E1\M$=M1$
80!"DO YOU WISH TO SEE THE CURRENT SCHEDULE? "\Y$=INCHAR$(0)
90IF Y$<>"Y" THEN GOTO 190
94J=0
96!" Date for this schedule is  ",M$," ",E
100!" Time",TAB(10),"Monday",TAB(20),"Tuesday",TAB(30),"Wednesday",TAB(40),
110!"Thursday",TAB(50),"Friday",TAB(60),"Saturday",TAB(70),"Sunday"
120FOR I=J TO J+15
130  !S$(I*80+1,I*80+78)
140  NEXT
160J=J+16
170IF J =48 THEN 190
180INPUT "  HIT RETURN FOR MORE",R$\GOTO 96
190!"DO YOU HAVE CHANGES TO MAKE?  "\R$=INCHAR$(0)
200IF R$<>"Y" THEN GOTO 350
210!"  TO ENTER NEW APPOINTMENTS FIRST CHOOSE DAY,D "
220!"        MONDAY=1; TUESDAY=2,.....SUNDAY=7"
240!"  ENTER TIME AS 3 OR 4 DIGITS IN 15 MINUTE INTERVALS BETWEEN"
250!"        900(AM) AND 800(PM)"
260!"ENTER NUMBER OF QUARTER HOURS INTERVALS,Q  "
270!"ENTER APPOINTMENT DESCRIPTION < 10 CHARACTERS,A ",A$
275!TAB(20),"D,T   ,Q,appointment
280INPUT "IN THAT ORDER NOW:  ",D,T,Q,A$
290IF T < 900 THEN T = T-40*INT(T/100)+720 ELSE T=T-40*INT(T/100)
300L = (T-540)/15 \ B = L*80 + D*10 +1
310FOR I = 1 TO Q
320    S$(B,B+9) = A$
325B = B + 80
330    NEXT
340GOTO 190
350!"  WANT ANOTHER DISPLAY?  "\Y$=INCHAR$(0)
360IF Y$ = "Y" THEN 94
370!"  WANT A PRINTOUT?  "\Y$=INCHAR$(0)
380IF Y$ <> "Y" THEN GOTO 460
390!#1 "     S   C   H   E   D   U   L   E "\!#1
400!#1 "              ",T$,"                Week  of  ",M$,"  ",E\!#1
404!#1TAB(15),E,TAB(25),E+1,TAB(35),E+2,TAB(45),E+3,TAB(55),E+4,
406!#1TAB(65),E+5,TAB(73),E+6\!#1
410!#1 "  Time",TAB(10),"Monday",TAB(20),"Tuesday",TAB(30),"Wednesday",TAB(40),
420!#1 "Thursday",TAB(50),"Friday",TAB(60),"Saturday",TAB(70),"Sunday"
430FOR I = 0 TO 47
440!#1 S$(I*80+1,(I+1)*80)
450   NEXT
460!"WANT TO STORE THIS SCHEDULE?  "\Y$=INCHAR$(0)
465CLOSE #1
470IF Y$ <> "Y" THEN GOTO 500
475OPEN#2, "SCHD" + T$ + ",2"
480WRITE #2,S$,M$,E
490CLOSE #1
500CHAIN "TK"
```

A$, is to be inserted here. However, we want to repeat this information in this column for as many quarter-hour periods, Q, as you have indicated. Thus, an appointment lasting three-quarters of an hour should appear three times on the schedule. A loop does this (310-330) and it is repeated once for each quarter-hour interval specified (310). The appointment information supplied, A$, replaces the current substring starting at the byte address, B, and continuing to B+9 (320). After each execution, B is advanced to the next line in the same column by adding 80 to it (325).

The method used to enter a new appointment is also used to delete a cancelled appointment. When asked for appointment information, you specify D, T and Q as they pertain to the cancelled appointment. Now, instead of supplying an appointment description, you enter enough blanks or spaces to cancel out the previous description.

The same entry technique is used to put times in the "time" column on the right lines. It can really be handled like any other appointment message. Since we have called the days of the week 1 through 7, it should be clear that the label for time column is 0. So, to put the label 11:00 in the time column, I respond to the prompt (280) with:

```
0, 1100, 1,      11:00
```

Notice the three blanks before 11:00 to indent it properly on the printout.

Flow Chart

A flow chart for the overall program appears as Figure 1. On each box are the numbers corresponding to the line numbers of the Program Listing. (The box number appears in brackets in the text which follows to identify the box being discussed; line numbers still appear within parentheses in the text.)

After the remark in the program to document it (10, 20), the first action requests you to identify the two files which will participate in producing and recording the schedule: The *from* file is a source of the schedule to be displayed or printed. The *to* file identifies where the revised schedule will be stored. You identify both of the files by striking a single letter, P, A, B or C. North Star Basic has a feature whereby a character entered at the keyboard is identified as a single input character string. No carriage return is needed — the

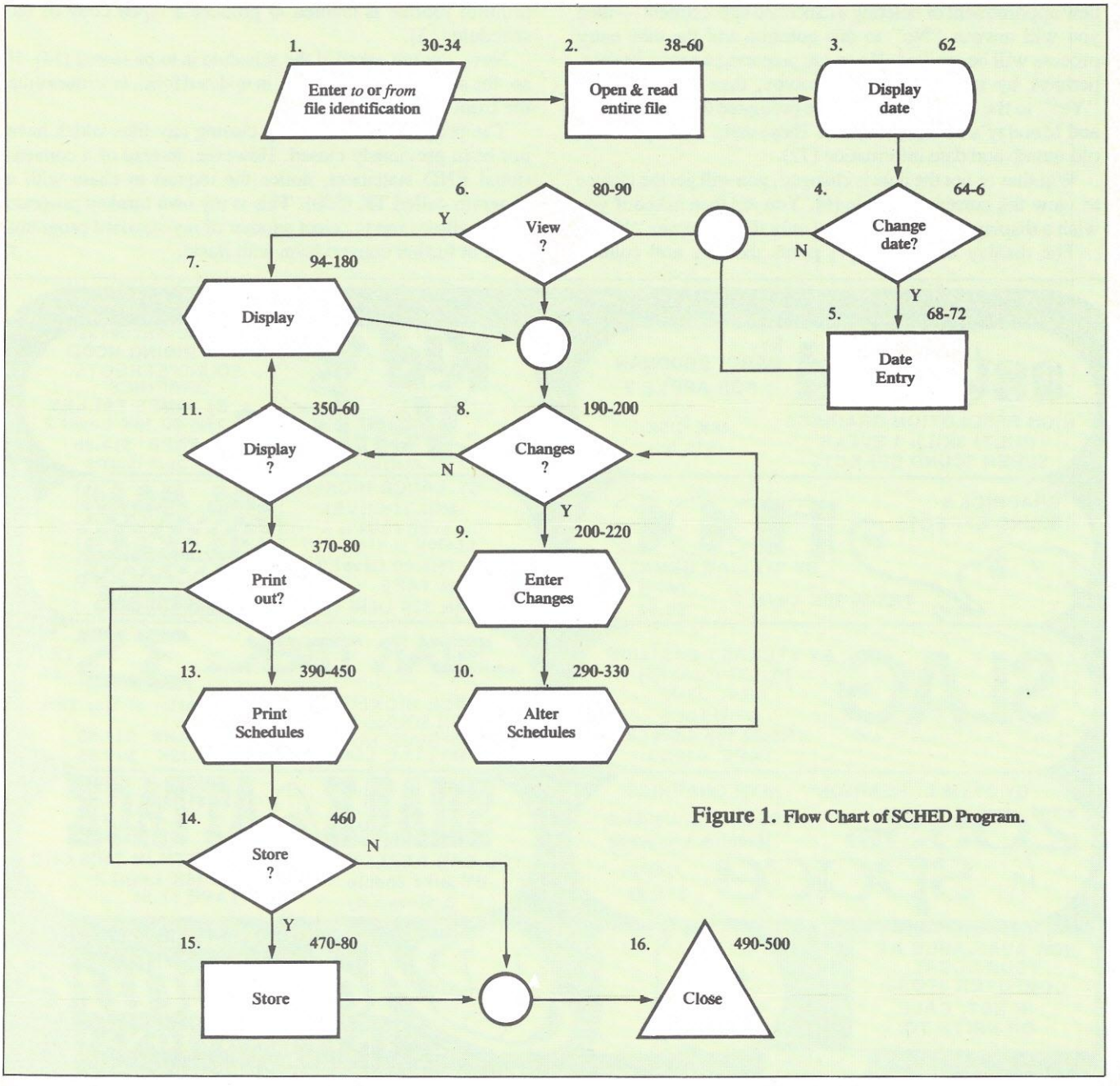


Figure 1. Flow Chart of SCHED Program.

program continues as soon as the key is struck. The console has the address 0. The function, INCHAR$ (0), absorbs the character represented by the key you pressed and identifies it with the character string named on the left of the equal sign. Thus in (32) F$ is used to contain the suffix of the from file — this letter is then printed. Then the suffix for the to file is obtained and identified with T$ [1].

The next action is to open and read the from file [2]. All four files have, as the first four letters of their name, SCHD. The next letter is P, A, B or C according to which of the four files is chosen. North Star Basic allows us to concatenate a fixed string with a variable string using + to indicate concatenation. If you indicate that the from file is the prototype file with the suffix P, then (38) will open the file, called SCHDP,2, where the 2 indicates that it is found on the second disk drive. The entire file will then be read (60) to produce the three variables, S$, M$ and E for the schedule proper, the month and the day. Next, the month and the day are displayed for you to verify [3].

Now you are asked if you wish to change the date. If you are making a change in an existing schedule — entering a new appointment or deleting a cancelled appointment — then you will answer "No" to this question and the date entry process will be omitted. If you are preparing a new schedule, perhaps by modifying the prototype, then you'll answer "Yes" to the date question and be prompted to enter a month and Monday's date [5] (68, 70). These dates will replace the old month and date information (72).

Whether or not the date is changed, you will get the chance to view the current schedule [6]. You are then asked if you wish a display which is provided only if you answer "Yes".

The display subroutine [7] prints the date and column headings and 16 lines or four hours of the schedule. To get the next four hours, you must hit the carriage return (180). The schedule is broken up into three displays to provide proper column headings for each.

After the display, if there is one, you are given the opportunity to make changes [8]. Each change — addition or deletion — consists of entering the four items as described earlier. You are prompted to enter these changes correctly [9].

After each change is entered, the schedule string, S$, is altered. The day, time and the number of quarter-hour intervals (D, T and Q) are used to find the byte position in the string where the appointment information, A$, will be entered [10]. You are then given a chance to make an additional change [8]. If you accept, it will be entered [9, 10]. If you decline, you are offered an opportunity to see the schedule display again [11]. If you accept a display, the altered schedule is presented in three parts as before [7], and then you get an opportunity to enter still more changes [8].

After declining further changes [8] and a further display [11], you forgo any further opportunity to make changes. At this point you are asked if a printout is desired [12]. If so, the printout routine is entered to produce a typed copy of the schedule [13].

Now you are asked if the schedule is to be stored [14]. If so, the schedule information, in updated form, is written onto the from file [15].

The program terminates by closing any files which have not been previously closed. However, instead of a conventional END statement, notice the request to chain with a program called TK (500). This is my own turnkey program which allows me to select another of my standard programs without further conversation with Basic. □

Number Converter

BY W.B. GOLDSMITH, JR.

What would your house number be if it were expressed in a number system other than decimal? What would be the decimal equivalent of your name if the letters that spell it were digits in some other number system? On a more practical note, wouldn't it be nice to be able to convert decimal to binary and hexadecimal without a pencil and paper?

Number Converter does it. You can now convert any decimal number to its equivalent in any integer number system from base 2 to base 32. You can also convert numbers in other systems to their decimal equivalents with a few keystrokes of your personal computer keyboard.

Once in awhile, I like to dig into the inner sanctum of my personal computer and write a short routine in machine code. Since I don't do this very often, I'm not conversant with the mental gymnastics needed to perform hexadecimal to decimal number conversions. Because it's an infrequent thing, I don't want to purchase one of those pocket electronic conversion calculators, and every time I find a handy-dandy look-up chart that will solve my number conversion problems, I lose the thing. Number Converter is a Basic program that cost a few hours to develop. It does the conversions that I need now and then and it answers those questions that don't really need answers such as: What is my telephone number in base 27?

If you need a hex, octal, binary, decimal or whatever number changer, this program is for you. When your children's math teacher starts addressing different number systems, you'll be glad you spent a short keypunch session with this one too. The program has some limits that depend pretty much on

A regular PC contributor, Mr. Goldsmith frequently writes articles on business applications he's developed for his SWTP system. His previous articles include "Installment Sales" (August 1980), "Reminder Lists" (September 1980) and "Income Statements (October 1980).

Program Listing

```
0010 REM ********************************
0020 REM *        NUMBER CONVERTER      *
0030 REM ********************************
0040 REM *      COPYRIGHT 1980 BY:      *
0050 REM *     W. B. GOLDSMITH, JR.     *
0060 REM *    LAKEWOOD, CALIF. 90712    *
0070 REM ********************************
0080 REM
0090 DIM D$(20),D(21)
0100 PRINT " NUMBER CONVERTER"
0110 PRINT
0120 PRINT "DO YOU WISH TO CONVERT 'TO' OR"
0130 INPUT "'FROM' THE DECIMAL SYSTEM",Z$
0140 IF LEFT$(Z$,1)="T" THEN 3000
0150 IF LEFT$(Z$,1)="F" THEN 200
0160 PRINT
0170 PRINT "SORRY, I DON'T UNDERSTAND YOUR"
0180 PRINT "ANSWER '";Z$;"'.  TRY AGAIN."
0190 GOTO 110
0200 REM ** DECIMAL TO OTHER BASE CONVERSION ******
0210 INPUT "WHAT DECIMAL NUMBER",N
0220 IF N>999999 THEN 2000
0230 PRINT
0240 PRINT "WHAT BASE SYSTEM DO YOU WISH TO"
0250 PRINT "CONVERT TO (PLEASE ENTER THE"
0260 INPUT " NUMBER)",B
0270 GOSUB 600
0300 REM ** CONVERSION CALCULATION ***************
0310  D(1)=N
0320 FOR X=1 TO 20
0330  DO=INT(B*((D(X)/B)-INT(D(X)/B))+.5)
0340 GOSUB 1000
0350  D$(X)=D$
0360  D(X+1)=(D(X)-DO)/B
0370 NEXT X
0400 REM ** OUTPUT ROUTINE ************************
0410 PRINT N;" EXPRESSED IN BASE ";B;"IS:"
0420 PRINT : Y=0
0430 FOR X=20 TO 1 STEP -1
0440 IF Y>0 THEN 460
0450 IF D$(X)="0" THEN 470
0460 PRINT D$(X);:Y=Y+1
0470 NEXT X
0480 PRINT :PRINT
0490 INPUT "WANT TO CONVERT ANOTHER",Z$
0500 IF LEFT$(Z$,1)="Y" THEN 110
0510 PRINT "O.K.,SEE YOU LATER."
0520 END
0600 REM ** SUBROUTINE TO CHECK BASE LIMITS *******
0610 IF B>32 THEN 2000
0620 IF B<>INT(B) THEN 2000
0630 IF B<2 THEN 2000
0640 RETURN
1000 REM ** DIGIT CONVERSION ROUTINE **************
1010 IF DO<10 THEN D$=STR$(DO)
1020 IF DO=10 THEN D$="A"
1030 IF DO=11 THEN D$="B"
1040 IF DO=12 THEN D$="C"
1050 IF DO=13 THEN D$="D"
1060 IF DO=14 THEN D$="E"
1070 IF DO=15 THEN D$="F"
1080 IF DO=16 THEN D$="G"
```

the number-crunching capabilities of your Basic interpreter, but it'll do the job for numbers in ranges we usually need.

User's Notes

Number Converter is an easy program to use. All required entries are prompted with explanatory questions and error trapping routines keep your answers honest. When you perform a conversion that stretches the limits of the computer math routines, you'll get a warning that the answer may not be exact. One nice feature of the math routine errors is that you can find them. If you convert from decimal to some other system (base 25, for example) then make the reverse conversion (from 25 back to decimal) you'll see any errors. If you don't see an error in the turnaround conversion, you can be fairly sure that neither conversion is in error.

Because of the general nature of the conversion attempts, you can't go from one system to another without making a decimal conversion along the way. If you need to go from base 27 to base 19, you'll have to make two conversions — base 27 to decimal, then decimal to base 19.

Programming Notes

Number Converter is written in SWTP 8K Basic Version 2.0. For the most part, it's a "plain vanilla" program that should fit most other Basics that have matrix variable capability and can handle character string variables. In a few places, I've used the multiple-statement-per-line feature, but it's only critical in one spot (Line 3110) and you can work around that if your Basic doesn't like more than one command on a line.

Some of the conversion routines use a bit of brute force. If you like more purity in your mathematics, you'll want to change some of the number crunching. But that's part of the fun in home computing — improving every program you can get your hands on. Please refer to the Program Listing as we tiptoe through the Basic.

The REM's in lines 10 through 80 make up the title block but don't affect program operation. The variable dimensioning statement of line 90 is necessary to allow your computer to reserve the required variable memory space. If you want to expand the number of digits addressed by the "Decimal to Other" routines, you'll need to change the DIM statement. D$ should be equal to the number of digits pos-

```
1090 IF DO=17 THEN D$="H"
1100 IF DO=18 THEN D$="J"
1110 IF DO=19 THEN D$="K"
1120 IF DO=20 THEN D$="L"
1130 IF DO=21 THEN D$="M"
1140 IF DO=22 THEN D$="N"
1150 IF DO=23 THEN D$="P"
1160 IF DO=24 THEN D$="Q"
1170 IF DO=25 THEN D$="R"
1180 IF DO=26 THEN D$="S"
1190 IF DO=27 THEN D$="T"
1200 IF DO=28 THEN D$="U"
1210 IF DO=29 THEN D$="V"
1220 IF DO=30 THEN D$="W"
1230 IF DO=31 THEN D$="X"
1240 IF DO>31 THEN 2000
1250 RETURN
2000 REM ** OOPS! THE NUMBER IS TOO LARGE *********
2010 PRINT
2020 PRINT "******** OUCH ********"
2030 PRINT "YOU TURKEY! YOU HAVE GIVEN ME"
2040 PRINT "TOO LARGE A NUMBER.  MY MATH"
2050 PRINT "ROUTINES WILL ONLY ALLOW ME TO"
2060 PRINT "WORK WITH DECIMAL NUMBERS OF"
2070 PRINT "FEWER THAN SEVEN DIGITS,"
2080 PRINT "OTHER SYSTEMS UP TO BASE 32, "
2090 PRINT "AND ONLY WITH INTEGER BASES"
2100 PRINT "LARGER THAN ONE."
2110 END
2990 REM ** CONVERSION TO A DECIMAL NUMBER ********
3000 PRINT "WHAT NUMBER DO YOU WISH TO"
3010 INPUT "CONVERT TO DECIMAL NOTATION",N$
3020 PRINT
3030 PRINT "IN WHAT BASE SYSTEM IS ";N$
3040 INPUT "(ENTER BASE NUMBER IN DECIMAL)",B
3050 GOSUB 600
3060  D=0
3070 FOR X=LEN(N$) TO 1 STEP -1
3080  F=0
3090 FOR Y=1 TO B
3100 READ A$
3110 IF MID$(N$,X,1)=A$ THEN DO=Y-1:F=1
3120 NEXT Y
3130 IF F=0 THEN 4000
3140  P=LEN(N$)-X
3150  DO=INT(DO*(B↑P)+.5)
3160  D=D+DO
3170 RESTORE
3180 NEXT X
3190 PRINT "THE DECIMAL EQUIVALENT OF ";N$
3200 PRINT "IN BASE ";B;"IS:"
3210 PRINT :PRINT D
3220 PRINT
3230 IF LEN(STR$(D))>5 GOSUB 3300
3240 INPUT "WANT TO CONVERT ANOTHER",Z$
3250 IF LEFT$(Z$,1)="Y" THEN 110
3260 END
3300 REM ** UNRELIABLE ANSWER CAUTION *************
3310 PRINT "********** CAUTION **********"
3320 PRINT
3330 PRINT "CONVERSIONS OF LARGE NUMBERS"
3340 PRINT "MAY BE INACCURATE IN THE LEAST"
3350 PRINT "SIGNIFICANT DIGITS DUE TO LIMITS"
3360 PRINT "IN MY MATH ROUTINES."
3370 PRINT "********** CAUTION **********"
3380 PRINT
3390 RETURN
4000 REM ** DUMMY! YOU GAVE ME THE WRONG BASE *****
4010 PRINT "******** OW! ********"
4020 PRINT
4030 PRINT "THE NUMBER YOU GAVE ME DOESN'T"
4040 PRINT "FIT THE BASE YOU GAVE ME!"
4050 PRINT
4060 PRINT "TRY AGAIN...TURKEY!"
4070 END
9000 DATA 0,1,2,3,4,5,6,7,8,9
9010 DATA A,B,C,D,E,F,G,H,J,K
9020 DATA L,M,N,P,Q,R,S,T,U,V
9030 DATA W,X
```

sible with a conversion (this dimension is usually measured by the largest binary word you'll deal with). D (n) must be dimensioned to the number of digits plus one. The extra position is calculated but not used in one of the FOR/NEXT exercises.

Statements 100 and 110 provide a title for your terminal when you run Number Converter. Line 110 is a little more than an innocent PRINT, and is the target for three GOTO's, so if you omit this line, change the GOTO of lines 190, 500 and 3250.

Lines 120 through 190 provide the operator a choice between converting "From" a digital number or "To" one. Although the error catcher in lines 160 to 190 may seem like frosting, I've found myself trying to enter numbers when the program wants a "To" or "From".

The statements in lines 200 through 270 provide the entry opportunity for the "Decimal to Other" conversion. The error trap in line 220 is based on accuracy limitations of the math routines in my Basic. If your machine can handle more than six decimal digits throughout the conversion process, you may wish to expand this limit or leave it out.

The subroutine at 600 to 640, which is targeted by line 270, is another error trapper. This little jewel insures that the base you've picked is an integer larger than one and no larger than thirty-two. The base 32 upper limit was picked somewhat arbitrarily, so you can change it to what ever you're happiest with. Thirty-two was picked because it's a multiple of sixteen, the hexadecimal base popular with computers, and because it's small enough to allow single character representation within the normal alphabetic notation. I also picked the notation scheme arbitrarily — digits that represent numbers from 10 to 31 are the standard alphabet less "I" and "O". If there's a standard convention on this, I'm unaware of it.

The base limit checking subroutine sends errant program runs to the "Turkey" message at line 2000. When you receive this slap on the wrist from your computer, you'll have to type RUN again. The error message terminates the program for two reasons. First, it causes you to start over as a penance for entering an illegal base number. Second, you may get to the error message from the middle of a FOR/NEXT loop. There's no graceful recovery from the failed loop arithmetic and trying to recover may leave your Basic stack pointer up a tree. Better to start over.

The actual "Decimal to Other" calculation is handled in statements 300 to 370. The computation is carried out to twenty digits by a successive divide and conquer FOR/NEXT loop. The original decimal number is divided by the base number. Any fraction that remains is multiplied by the base to determine the value of the least significant digit. The digit is given a label by subroutine 1000 and stored in D$(X), where D$(1) is the least significant digit and D$(20) the most significant digit. The remaining integer number is again divided by the base with the new fraction used to determine the value of the next most significant digit (D$(2)). The process is repeated until all digits have been labelled.

Usually, the decimal number won't have twenty significant digits in the new base. We'll take care of all those leading zeroes later. Line 360 is the reason we dimensioned variable D (X) for twenty-one slots instead of the twenty digits. During each pass through the FOR/NEXT, line 360 calculates the next D (X) term. It was more economical to "waste" a slot for D (21) than to write three or four more lines of code to keep from defining a D (21).

You can convert from decimal notation to any of thirty different number bases and back.

The output routine in 400 through 470 uses another FOR/NEXT to spill the matrix of digits onto the system terminal. In this case, the most significant digit is in D(20) so we step the loop down from 20 to 1. The IF D$(X)="0" in 450 is one-half the leading zero suppression device. The Y=0 in 420 sets a flag that is tested by 440 and incremented when the first non-zero digit is printed by 460. After the flag is incremented, line 450 is bypassed. (If not for the flag, *all* zeroes would be suppressed.) Multiple statements per line are used in three places in this output routine. If your Basic won't handle multiple statements on a line, you can move the second statement to an intermediate line. For example:

```
0460 PRINT D$(X);
0465 Y=Y+1
0470 NEXT X
```

The rest of the logic will still work without a hitch. I used the feature to sneak in a few statements after I had my program numbered. By stacking multiple statements on a line, I was able to keep my line numbers ending in zero.

Statements 490 through 520 provide a wrap-up for this half of the program. If you want to keep converting numbers, answer "Yes" to the "WANT TO CONVERT ANOTHER?" question and the program will recycle from the top.

Conversion to a decimal number is a shorter process. The abbreviation is due to a different technique for matching the digit values and a more straightforward arithmetic.

The "Other to Decimal" conversion starts with the entry scheme of lines 3000 to 3050. The base number error checking subroutine of 600 to 640 is used again to keep things on an even keel.

A set of nested FOR/NEXT loops handles all of the computation. The "X" loop of 3070 to 3180 provides a major cycle through each digit of the base number (least significant digit first). The "Y" loop compares each digit with DATA values to determine the decimal numeric value of the digit. An error checking feature is built into steps 3080, 3110 and 3130 that depends on the multiple statement per line feature of my Basic. If your Basic doesn't have this capability, you'll have to add something like:

```
3115 IF MID$(N$,X,1)=A THEN F=1
```

Completing any of the "Y" loops with F=0 indicates that no match was found for the base number digit. If this happens, you'll get the "Dummy" message of 4000 to 4060 and find the program terminated. Specifying a digit larger than the base (for example, a "9" for a base 8 number) will guarantee you a "Dummy" name calling session with your computer.

Normally, if things go right, and after a match is found, the base number is raised to the power of its positional notation and multiplied by the decimal value of the digit in that position. After the exponenting and multiplying in steps 3140 and 3150, the decimal values are summed in 3160.

The RESTORE statement of line 3170 is critical to reset the Basic data pointer for the next READ data sequence.

Because exponentiation is a little less precise in my Basic than other arithme-

tic functions, answers of greater than five decimal digits may not be accurate. The accuracy deviation will be in the least significant digits, but if you're trying to calculate a monster jump address for a machine language program, one digit in error is a disaster. Line 3230 checks for decimal results of greater than five digits. If a suspect answer is given to you, the warning message issued by the subroutine at 3300 will accompany the answer. As we mentioned earlier, you can check for accuracy by reconverting the answer to its original form with the other half of the program. If the program is not giving you an accurate answer and you need one, try breaking the number into smaller pieces and adding the answers.

Lines 3240, 3250 and 3260 look strangely like 490, 500 and 520. They should because both sets of statements let you choose to convert another number or stop for now. You could save some memory by changing 3240 to "GOTO 490" and deleting 3250 and 3260. The extra GOTO won't strain your computer's capacity and the possible confusion in your listing shouldn't be a big problem.

That's it. Number Converter will allow you to convert from decimal notation to any of thirty different number bases and back. While the greatest use for this program will probably be in converting from hexadecimal to decimal (and vice versa), its greatest fun will be in converting to base 7 or base 23. It is also a great training aid for studies in number bases. Best of all, the price is right. In less than an hour, you should have Number Converter keypunched into your personal computer and be changing bases to your heart's content. □

Sample Run

```
READY
#RUN
NUMBER CONVERTER

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? FROM
WHAT DECIMAL NUMBER? 4724

WHAT BASE SYSTEM DO YOU WISH TO
CONVERT TO (PLEASE ENTER THE
NUMBER)? 27
4724  EXPRESSED IN BASE 27 IS:

6CS

WANT TO CONVERT ANOTHER? Y

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? TO
WHAT NUMBER DO YOU WISH TO
CONVERT TO DECIMAL NOTATION? 6CS

IN WHAT BASE SYSTEM IS 6CS
(ENTER BASE NUMBER IN DECIMAL)? 27
THE DECIMAL EQUIVALENT OF 6CS
IN BASE 27 IS:

4724

WANT TO CONVERT ANOTHER? Y

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? TO
WHAT NUMBER DO YOU WISH TO
CONVERT TO DECIMAL NOTATION? BUD

IN WHAT BASE SYSTEM IS BUD
(ENTER BASE NUMBER IN DECIMAL)? 32
THE DECIMAL EQUIVALENT OF BUD
IN BASE 32 IS:

12173

WANT TO CONVERT ANOTHER? YES

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? FROM
WHAT DECIMAL NUMBER? 12173

WHAT BASE SYSTEM DO YOU WISH TO
CONVERT TO (PLEASE ENTER THE
NUMBER)? 32
12173  EXPRESSED IN BASE 32 IS:

BUD

WANT TO CONVERT ANOTHER? YUP

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? TO
WHAT NUMBER DO YOU WISH TO
CONVERT TO DECIMAL NOTATION? 7FFF

IN WHAT BASE SYSTEM IS 7FFF
(ENTER BASE NUMBER IN DECIMAL)? 16
THE DECIMAL EQUIVALENT OF 7FFF
IN BASE 16 IS:

32767

WANT TO CONVERT ANOTHER? Y

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? FROM
WHAT DECIMAL NUMBER? 32767

WHAT BASE SYSTEM DO YOU WISH TO
CONVERT TO (PLEASE ENTER THE
NUMBER)? 2
32767  EXPRESSED IN BASE 2 IS:

111111111111111

WANT TO CONVERT ANOTHER? Y

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? TO
WHAT NUMBER DO YOU WISH TO
CONVERT TO DECIMAL NOTATION? 10101010101010

IN WHAT BASE SYSTEM IS 10101010101010
(ENTER BASE NUMBER IN DECIMAL)? 2
THE DECIMAL EQUIVALENT OF 10101010101010
IN BASE 2 IS:

10922

WANT TO CONVERT ANOTHER? YES

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? FROM
WHAT DECIMAL NUMBER? 10922

WHAT BASE SYSTEM DO YOU WISH TO
CONVERT TO (PLEASE ENTER THE
NUMBER)? 8
10922  EXPRESSED IN BASE 8 IS:

25252

WANT TO CONVERT ANOTHER? Y

DO YOU WISH TO CONVERT 'TO' OR
'FROM' THE DECIMAL SYSTEM? FROM
WHAT DECIMAL NUMBER? 10922

WHAT BASE SYSTEM DO YOU WISH TO
CONVERT TO (PLEASE ENTER THE
NUMBER)? 16
10922  EXPRESSED IN BASE 16 IS:

2AAA

WANT TO CONVERT ANOTHER? NOPE
O.K.,SEE YOU LATER.
```

Microbicide

BY DAVID LUBAR

You finally enter the last line of a program and, eager to fight the galactic interlopers, start a RUN. The alien saucer approaches. You hit the key which moves your ship to the left. Nothing happens. You hit another key. The program dies.

If this has never happened to you, you're either very lucky or a minor deity. Mortals have to cope with bugs. Finding and fixing such problems is part art and part science. The following article, while not exhaustive, discusses techniques for debugging programs.

I'll cover two types of programs: your own, and those taken from listings. In most cases, listings will be error free. Occasionally, an error will creep in and usually the problem comes from a typing mistake. Before entering a listing, it is a good idea to look over the program, following the logic flow. Try to determine what functions are performed by each section. When entering your own programs, make sure you have a neat copy of *exactly* what will go into memory.

To fix a bug, you have to know where it occurs. If there is an actual error, the error message will give the line number where the program stopped. In this case, LIST the line and compare it to the written listing. If there is no mistake, work backwards. Find the lines that assign values to any variables in the problem line. The approach from this point overlaps the techniques that you'd use if the bug didn't stop the program.

An understanding of the logic of the program will help narrow doown the area of the bug. Let's take an example. Suppose a program was designed, in part, to move a figure whenever a key is pressed. One part of the program gets the data from the keyboard; another part actually moves the figure. The problem could be in either part. Since the input portion comes first, we'll start with that. When this program was written, it was assumed that certain variables would have certain specific values, depending on which key was pressed. First, find out if the variables have the correct value.

One way is to use the DSP ("Display") function, available in Apple Integer Basic. Used either in a program line or in the direct mode, this function prints a variable on the screen whenever the variable gets a new value. The line number is also displayed. In our example, you could put a line at the end of the key-read routine, looping back to the beginning of the routine. RUN the program and press the keys, keeping an eye on the values of the variables involved.

If your Basic doesn't have DSP command, there is another approach. Rather than turning the key-read routine into a loop, add a line which stops it after one pass. After a RUN, enter the direct-mode command PRINT A, where A is the variable used in the routine. If the value returned isn't what you expected, the problem has been found. If the value is correct, go on to the actual move routine (after deleting the extra line that was used to stop the key-read portion). Again, check the values of all variables involved. If the values are correct, the problem probably concerns logic flow. (In other words, if the program gets the key value but doesn't go to the move routine, nothing will happen.)

Check all conditional branches. Those that contain a lot of ANDs, ORs or NOTs should be examined to determine what happens in all possible cases. This can be done in the direct mode. For example, suppose the line is:

```
10 IF X=5 AND (Y<>7 OR Z=3) THEN 100
```

Give values to the variables, then enter the command:

```
PRINT X=5 AND (Y<>7 OR Z=3)
```

This will return 0 if the statement is false, and 1 if it is true. The branch to 100 will occur only when the value is 1. If the branch doesn't occur at the right time, something is wrong with the logic. Try rewriting it in a different form.

(Not all computers follow the same conventions in the values returned for logical true and false statements. The values of 0 and 1 for false and true hold for both Applesoft and for Apple Integer Basic. They also hold on the Pet, though the Pet can take things a step farther. Pet will interpret "5 AND 4" as producing 4, which is the result of ANDing 1000 with 1001 in binary. In TRS-80 Level II Basic, –1 (minus one) stands for a logical true and 0 (zero) for a logical false. Check to see what convention your computer uses.)

Pay close attention to the difference between AND and OR, especially when used in combination. If the line is complex, break it up into several statements. Reducing NOTs can make the line easier to follow. For example, NOT A AND NOT B can be replaced with NOT (A OR B). You can find a full list of equivalent expressions in any book on elementary logic.

If the variables don't have the correct values, you will have to step through the program, acting as a computer. Assume a certain input, then see what values the lines give the variables. It is easy for errors to slip in when using certain functions. For example, INT truncates a number. If you wanted to round off a

number, INT could give the wrong result. Make sure you have followed the correct rules for precedence (multiplication is performed before addition). When in doubt, us parentheses. If there is a loop, any initialization of variables should be done before the loop, except when using the loop to initialize arrays. For example:

```
10 X=0
20 FOR I=1 TO N
30 X=X+1
40 NEXT I
```

In this case, placing line 10 inside the loop would result in a value of 1 for X no matter what the value of N was.

Typographical errors can be difficult to catch, especially when they don't cause syntax errors. Suppose the correct line is:

10 IF X=3 OR X=7 OR X=11 THEN 100

If the shift key isn't used for the last =, the line will become:

10 IF X=3 OR X=7 OR X–11 THEN 100

In the first case, the branch to 100 will only occur when X equals one of the three listed values. In the second case, the branch will always occur unless X=11. (When X=11, X-11 will have a value of zero. In any other case, X-11 will be non-zero, and the condition will be interpreted as being true.) This is the type of error that can keep you searching for hours.

Passing mention should be given to the TRACE command. Often, this is of little help since it fills the screen with numbers at a rapid rate. It can be used to determine whether a branch is being performed. The best approach is to stop the program and scan the numbers on the screen, looking for the line number of the branch command. Once you've found that, you can see where the program jumped.

Also, you can use TRACE and its companion NOTRACE as a numbered program statement — for example, 30 TRACE . . . 60 NOTRACE. In this case, the computer will begin TRACEing only when it encounters line 30, and will stop TRACEing when it encounters line 60. This lets you trace only the part of a program where you suspect the bug is.

(Again, conventions vary from machine to machine. TRACE is the command on the Apple; TRS-80 Level II Basic's equivalent is TRON. Check your manual to see if this function is available in your Basic and what the proper syntax is.)

In general, debugging consists of the following step:

1. Become familiar with the logic of the program.
2. Determine the areas that control the problem portion.
3. Work downward, starting with areas that pass on values or contain branches.
4. Determine whether variables have the expected values.
5. Check the logic of lines and make sure that the desired functions are being used.

If all else fails, SAVE the program and come back to it later. □

HARRY SHERSHOW — Dept. Editor

CSC Wins First Microcomputer-Chess Tournament

BY HARRY SHERSHOW

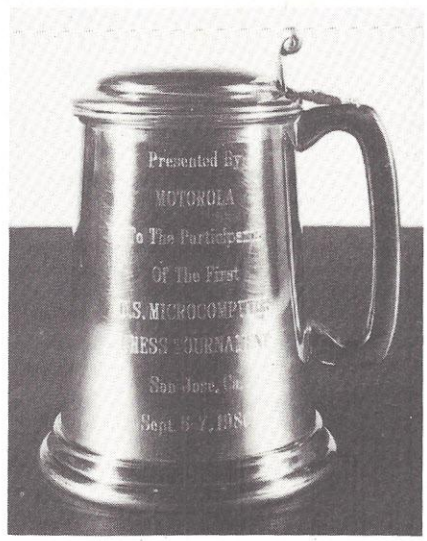
The first U.S. Microcomputer chess Tournament was co-sponsored by *Personal Computing*, Mychess Team, Applied Concepts and Motorola. The event was held at the LeBaron Hotel in San Jose, California, Sept. 5, 6 and 7, and was won by "CSC" (Champion Sensory Challenger, the newest chess product from Fidelity Electronics — but not yet on the market.) The winning unit was housed in a special attache case which, someone quipped, was handcuffed to the wrist of Mike Samole, VP of Fidelity, and which he carried in and out of the playing hall every day and night of the three-day affair. This heavy veil of mystery and security had other entrants swapping speculations. Meanwhile, questions from the approximately 500 spectators who came during the three-day tournament kept popping like corks from bottles of wine: "*Whose* program was in CSC?" "*Why* will it be unavailable in the market until some future date?" "How will eventual purchasers know that the CSC unit they buy, when it is finally on the store shelves, will be *identical* to this excellent tournament entry?"

The speculations mentioned rumored complex contract litigation, copyright controversies, chip availability, lawsuits, counter-suits, etc. The conclusions finally reached were that all these speculations and questions will probably be answered satisfactorily in three or four months and the chess world will finally have a Stand Alone Device that will play chess in the estimated level range of 1700-1800 in tournament time. So, let CSC come out of the shadows and sit on the throne until next year, when another warrior will appear to wrestle it for the crown.

This tournament was no snap for the Fidelity machine and turned out to be exciting and close, as organizer George Koltanowski had predicted. CSC had to win a couple of tough games from MYCHESS and BORIS to finish on top with its perfect 4-0 score (see final-result table.)

If Oscar-like titles, other than winner, had been bestowed, they would have gone to Tryom's Chess Champion Super System 3 for "The Most Beautiful Girl at the Show;" to Murray Lane's TC86 for "The Most Powerful Unit" (16-bit CPU, 9 MHz); to Atari for "The Most Sensible Unit," (just plug it into your TV); and to Applied Concepts' Modular Game System for "The Most Versatile Unit" (expandable and changeable by plugging new modules into small mainframe.) All the tournament games are being collected by Koltanowski and will appear in a small booklet which will be available free to those who request it. Details on how to obtain such a pamphlet (when it is ready) will be listed here. Annotated games of the tournament will be appearing here monthly together with discussions, interviews and general comments and events that were overheard, photographed or recorded.

At the closing-ceremonial dinner Saturday night, specially-engraved pewter mugs, (donated by co-sponsor Motorola,) were presented to all participants of the tournament. Also awarded were Motorola's cash prizes of $350 to Bill Fink (of the SFINK program) — the highest finishing non-commercial entry; and $250 to Murray Lane, the other non-commerical entrant (there were only two.) Murray's program failed to win a game and gamblers will argue that if the horse that finishes last in a race doesn't get part of the purse, why should last-place Murray Lane? At a pre-dinner discussion (involving philosophers, computer systems analysts, chess masters and a few kibitizers, all a little woozy from the delightful heady wine donated by the Paul Masson Wineries) a decision was reached. Because four other non-commercial entries had indicated they were going to enter the tournament but failed to show up why not give it to Murray who did have the courage to come. He might have beaten those other non-commercial programs and finished in third place. Also, Murray deserved some award for being the first person to bring a 16-bit CPU, 9 MHz clock, Single Board Computer, to any microcomputer chess tournament. (Visions of powerful units about to come into the market.)



Motorola donated these engraved pewter mugs which were awarded to all participants at the computer-chess tournament. In addition, to encourage development of chess programming, Motorola awarded two cash prizes ($350 and $250) to the two non-commercial entrants.

Motorola's interest in this tournament is in its 6800 series chips. The Motorola 6800 with 16K RAM and 2K ROM was in a homebrew unit built by Mike Johnson of the British Post Office. Two years ago, in London, at a microcomputer chess tournament there, that unit and its resident program of MIKE finished in first place.

Bryce Perry directed the tournament, (when he wasn't tripping over all the extension cords) and the organizer was George Koltanowski. A chess-tournament organizer is like a busy bee that searches through the garden for honey that will bring out other bees (chess players) and, in this case, make the chess tournament a success both in audience turnout and in board play. George has had a long romance with

chess. He writes about chess (San Francisco Chronicle and syndication), he entertains at chess meetings, he talks about chess and, he admits, he dreams about it. He loves chess. He loves the people who play chess and, now, with his first exposure as organizer of a computer-chess tournament, a new romance may have come into George's life.

When he wanders around a chess hall and stops at a demonstration board to discuss what is going on, all eyes follow him and everything stops. Even the "keyboard operators" turn their ears in his direction while they continue to run the machines. George is probably the best chess-MC in the world. He magnetizes the audience with every word. When he occasionally sneezes some spectator will ask another, "What did he mean by that?" As much as George loves chess and chess players the love is returned. There is no knowledgeable chess player around who wouldn't trudge 10 miles or more on foot to listen to this kind, warm, humorous and brilliant man or watch him do his tricks with chess pieces (blindfold chess and knight's tour.)

George captures the audience's affections by his natural ability to communicate with them on their terms. He stops at a demonstration board of the CSC vs. MYCHESS game. "What do we have here?" he asks. "You can see if White should now move like this —" He stops, pulls his hand away from the chess piece he was about to move and turns to the audience. "I will ask you. What do *you* think White should do?" Some one suggests pushing a pawn. "OK," responds George. "Look! We move the pawn here. Black moves the knight here. White responds with his own knight here — he has no other move — and then the black bishop comes out and goodbye Mr. White. Anybody else?" "Move the rook to e5!" shouts someone. "A good move," says George. He puts the rook on e5. "But look here. Black comes out with his own rook and bishop must take. Then black puts his knight right here and we have mate in two moves!"

There has been no hesitation in George's delivery. No long think-time. Chess Masters don't think; they react. The answers and analyses came quickly and directly like someone reading a newspaper out loud. "Any more sug-

George Koltanowski, "Mastermind of The Chessboard," organized the microcomputer chess tournament. George is world-renowned for his blindfold chess demonstrations.

Final Results of Tournament

Finish	Player's Number	Program Name & Unit	Round 1			Round 2			Round 3			Round 4		Final Score
			Col.	Opp#	Score	Col.	Opp#	Score	Col.	Opp#	Score	Col.	Opp.#	
1	#10	"CSC" (Champion Sensory Challenger) On S.A.D.	W	#5	1	B	#1	2	W	#3	3	B	#9	4
2(Tie)	#1	MYCHESS "B" On Cromemco	B	#6	1	W	#10	1	B	#2	2	B	#3	2 1/2
2(Tie)	#2	BORIS "X" On S.A.D.	W	#7	1	B	#9	1 1/2	W	#1	1 1/2	B	#6	2 1/2
2(Tie)	#3	BORIS 2.5 On S.A.D.	B	#8	1	W	#4	2	B	#10	2	W	#1	2 1/2
2(Tie)	#9	CHESS CHAMPION SUPER SYSTEM 3 On S.A.D.	B	#4	1	W	#2	1 1/2	B	#7	2 1/2	W	#10	2 1/2
6	#5	Atari 4K "A" On TV Interface Unit	B	#10	0	W	#8	1	B	#4	1 1/2	W	#7	2
7(Tie)	#4	MYCHESS "A" On Cromemco	W	#9	0	B	#3	0	W	#5	1/2	B	#8	1 1/2
7(Tie)	#7	SFINKS On TRS-80 Disk	B	#2	0	W	#6	1	W	#9	1	B	#5	1 1/2
9	#6	ATARI 4K "B" On TV Interface Unit	W	#1	0	B	#7	0	B	#8	1	W	#2	1
10	#8	LANE'S TC'86 On S.A.D.	W	#3	0	B	#5	0	W	#6	0	W	#4	0

Cumulative scores after each round.
Usual 3-minute rule in effect.
S.A.D. = Stand Alone Device.

Organizer: George Koltanowski (assisted by John Urwin.)
Director: Bryce Perry.
Sponsors: PERSONAL COMPUTING, Applied Concepts, MYCHESS Team; Motorola

gestions?" The audience falls silent. He puts the pieces back in their original positions. "OK," he says, like a professor addressing a freshman class for the first itme. "Look what happens if we bring the white bishop over *here*!" He puts the white bishop there, then steps back to admire the new design. "Now, goodbye Mr. Black. He cannot avoid losing his rook like this —" he scatters pieces about — "and you can see for yourself."

Later, George sets up an end-game problem on an available demonstration board and announces to the spectators: "On a piece of paper write down your solution. This is a mate in three. We will put all the answers in a box and the first five correct answers we pull out will get a reward of a free copy of my book." When all the solutions are in the box, he calls upon an 11-year-old boy, who himself has submitted a solution. The young spectator keeps pulling out slips of paper until five correct solutions are found. George points out the solution and also demonstrates the wrong answers and explains why they were wrong.

This bantering between "Kolty" and the audience around the demonstration boards, is the one activity that makes a computer-chess tournament different from a human-chess tournament. When a human chess tournament is announced the people who come are all players. They bring their own chess pieces and their own boards with them. There is no seated audience watching. Computer chess tournaments, on the other hand, have finally become a spectator sport and one can expect future audiences to be larger. There will also be more interactivity between the spectators and the commentators.

Chatting With a Chess Master

George Koltanowski has one of the most amazing minds in the world. He is a human answer to the mechanical computer. He has played blindfold *chess against 34 simultaneous players without losing a game. When one considers that at every chessboard there are 10^{126} possible chess positions during a game, it is a stupendous feat of the human mind to play blindfold again 34 such boards all at the same time. No single computer with all its speed and all its huge chunks of memory could duplicate that feat today.*

When George grew tired from his constant wanderings about the hall, he sat down for a brief rest. This was a good time to have a chat with him.

"When did you learn to play chess?" we asked him.

"I learned when I was 14 from my father. The age of 14 is really a little too old to learn chess. As you get older you get other ideas, like girls, that distract you. But, don't get me wrong. It's never too late to learn. You can learn when you're 60 or older. In fact, for older people it's a marvelous game if you are retired and sitting around. People can play alone — especially with these computers. They don't need company. They don't have to travel miles to a chess club. They can even play chess by mail and make many friends that way."

"Why do you say 14 is a little too old to learn?"

"Because between the ages of six and nine a boy — or a girl — is an independent creature. And if you teach him the game at such an early age, he will continue to play the game for the rest of his life. A child takes to chess just like a duck takes to water. His mind is clear and his memory is free and he is hungry for knowledge."

"Is it difficult for older people to learn to play the game?"

"Difficult, yes. But not very difficult. You have to want to learn. Most people will say 'I don't have the brains to learn to play chess.' But you see, no one has ever proved you need 'brains' to play chess. Sure. To become a champion you have to be like Bobby Fischer. You have to have a sixth sense. What do we mean by 'brains?' We have thousands and thousands of people who dropped out of school at an early age but who now play chess at a high level. They can even beat college professors. So, playing chess has nothing at all to do with 'having brains.'"

"What about computers and chess?"

"A wonderful thing. Imagine having an instructor by your side all the time and any time? The machine can teach you how to play. It will tell you what move to make and why. The why is the important part of the learning process. Sure, there are computers out today that tell you what is the best move to make, based on the search tree. But they won't tell you why. Take Morphy, for example. He sacrificed a pawn early in the game to get an open file. And his opponents usually hung onto that pawn for dear life and only gave it back when they were about to be mated. It is because of Morphy that we say always accept a pawn sacrifice and always be willing to give it back if you have to. Thus you will play a better game. Computers, as teachers, will point this out. But to get to my point. At least by the age of nine youngsters should be taught by their parents to play chess. After that they start playing baseball and other things. Probably the age of nine is the best age to start teaching the youngster other things, like music, art, reading. You will probably find that all the people who became geniuses later in life, were taught as children by their parents at home. Apart from the school."

"Are computers the best tool for chess teaching?"

"No. The computers have many good merits. But they are not the whole solution. Otherwise, what am I doing here? Why couldn't you have a computer organize the tournament? But the whole point about computers and chess is that the computer gives the players a companion, or an opponent. This is the whole thing. You have to have someone to keep playing with so you can develop your

own ideas. And that is where computers come in nicely. In my own lifetime, I was champion of Belgium at the age of 17. I was champion for six times. I became a chess mentor about the time the Civil War broke out in Spain. I happened to be there as trainer for the chess team of the Madrid Athletic Club. When the war broke out I decided to become a chess professional and managed to get to New York."

"Have you ever done anything besides playing chess?"

"In 1941, I was a diamond cutter in New York. In 1945 I married and decided to give up diamond cutting and go on the road with chess. That was what I always wanted. And with encouragement from my wife I did it."

"What about these computers playing here at the tournament?"

"They play pretty good, especially in the middle game. Some have lousy opening programs and should be taken back to the drawing boards. One program I saw brought his Queen out on the third move and then for the next 5 or 6 moves kept moving it around in circles, wasting moves and giving the opponent a chance to develop. Of course it lost its game. After watching the games here I must assume that, most of all, the machines don't know how to play the endings well, yet."

"What do you think of this tournament we are running here?"

"I will have to wait until tomorrow when the tournament is over. But this is really a precedent. This is a whole lot different then the ACM tournament last year in Detroit with those big machines. Compared to the way this microcomputer chess tournament is run, it is far ahead of that ACM tournament. The hall here is beautiful, as you can see. The lighting is good. The hotel service is good. In Detroit they had nothing. It was in a cold, bare basement room."

"What about computers being allowed to play in human tournaments?"

"I am of the opinion that computers should combine with humans in chess play. Computers cannot be stopped. They are going to go ahead no matter what anyone thinks. They can't go backwards. If they stopped their development right now, they're already amazing. But they're going forwards all the time. And as chess players, the computers with their programs are 99% better than the ordinary chess player. The computer will undoubtedly become a companion to a chess player. Some of the machines here today are capable of doing better. Why are they doing worse than they're supposed to? I'm sure they'll take them back and work on them. And, after all, this tournament should serve as a place to test new programs as well as fight for the championship title. And we do have some experimental models here. I believe from what they have learned here, those experimental models will be greatly improved and will show up on the market next year and possibly be able to beat the winner of this tournament. Going ahead. Moving ahead. That's what is happening in the computer world, whether it's for chess or for guiding a missile.

"Our biggest problem in this tournament was that so many interesting machines said they were coming, but they never showed up. I would propose that in future tournaments all entrants would be declared two months in advance. Every entrant would have to send in $100 entry fee. This entry fee would then be given back when they appeared at the tournament. All in all this was a good beginning and a great success. The next one should be even better."

Up and Coming

Having heard that Tom Lucero, (4310 N. Spaulding. Apt 3, Chicago, IL 60618,) was participating in Varn Fields' Computer-Postal-Chess Tourney, we contacted him. The name of his program ("PIRANHA") indicated that he probably had a ferocious entry coming up. Tom's response:

"PIRANHA is a disk-based program in BASIC and miniassembler! The BASIC program is a driver/option opening routine. The machine code is 9K; the book is another 6K. The BASIC is 5K. I am currently rewriting the program in LISA assembly to add lines, graphics and to make changes possible. I have borrowed ideas from Peter Frey's 'Chess Skill in Man and Machines,' and from 'Sargon in Z-80' by Dan and Kathy Spracklen.

"My program uses both full-width and selective search. The selective search is a kludge fix for the horizon effect; so that it is able to look deeper at 'live' positions. An example of this is shown in the following to-date game in Tourney 80-C of Varn Fields' Tournament:

PIRANHA	SARGON-BORIS
1. d4 (book)	Nf6
2. c4 (book)	e6
3. Nf3 (book)	Nc6?

Now, theory recommends 4. Nc3. But, PIRANHA came up with 4. d5! It had examined 2,485 positions at up to 7 ply before settling on this. 4. Nc3 was rated second. A main-line analysis showed 4. d5 exd5 5. cxd5 Nb4 6. Nc3 Bc5 7. g3 . . .The analysis indicates that PIRANHA is a little pawn-pushy; but fundamentally its line is playable.

"PIRANHA has yet to beat me except at 5-minute chess. Participation in Varn Fields' tournament is its first competition. The program does not have time-control features, yet, as are used in tournament play. I am a 1744 USCF player and 1608 in USCF Postal (equal to 2050 over-the-board.) I would estimate PIRANHA at 1350 USCF and 850 USCF Postal. PIRANHA has yet to play a decent game. But wait until this little fish begins to fill its mouth with some sharp teeth!"

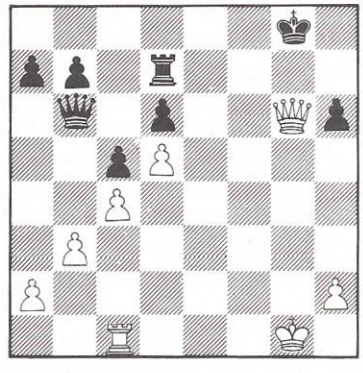
MYCHESS Shines

"During the grandiose Paul Masson Vineyards 8th Annual Open Air Class Championship," writes George Koltanowski, "780 players participated! They came from six foreign countries and 30 States. One lone microcomputer had the courage to enter the battle also. That entry was MYCHESS and in the simultaneous exhibition with Dr. Max Euwe, the computer had the Dutchman beat after Max had committed a blunder on his 33rd move. However, instead of trading a rook for Max's Queen, MYCHESS started getting fancy in the ending, knocking off idle pawns and wound up in a draw. On one day of the tournament, we had a Question and Answer period. Boris Spassky, who had participated in the demonstration, declared that he did not care to play against computers in exhibitions. (Someone had asked him why he had avoided playing against his namesake.) 'I'm just plain afraid of them,' said Boris (the man, not the machine.) But he did agree that they had come a long way. In 10 or 20 years, he believes, computers will be ranked among the top 'players' in the world.

"In explaining his game against MYCHESS, Dr. Max Euwe pointed out that most computers still were in need of great improvement. This was especially true, he said, in the endings. He has grave doubts that any of them will ever become World Champions. On the question whether computers should be allowed to play in Candidates Matches, Dr. Euwe said, curtly, 'We will worry about that when the time comes!'

"In the 21-board simultaneous exhibition by Dr. Euwe, he won 18, drew two and lost one. The MYCHESS game follows:

DR. EUWE WHITE	MYCHESS BLACK
1. d4	Nf6
2. c4	g6
3. g3	Nc6
4. d5	Ne5
5. Qd4	d6
6. Nc3	c5
7. Qh4	Bg7
8. Nf3	Nxf3ch
9. exf3	e5
10. Bh6	0-0!
11. Bxg7	Kxg7
12. Be2	Bf5
13. 0-0	Qb6
14. b3	Rad8
15. Rfe1	h6
16. f4	e4
17. g4	Qa5
18. Rac1	Bd7
19. g5	Ng8!



Position after 33. Qxg6 ch?

20. Bg4	Bxg4
21. Qxg4	f5
22. gxf6ch	Nxf6
23. Qg2	Nh5
24. Rxe4	Nxf4
25. Qg3	Nh5
26. Re7ch	Rf7
27. Rxf7ch	Kxf7
28. Qe3	Kg7!
29. Qe7ch	Kg8
30. Ne4	Qb6
31. Nf6ch	Nxf6
32. Qxf6	Rd7
33. Qxg6ch??	Rg7
34. Qf6 (Diag.)	Qb4
35. Rd1	Qa3
36. Re1	Qxa2
37. Re8ch	Kh7
38. Re7!	Qb1ch
39. Kg2	Rxe7
40. Qxd6	Rg7ch
41. Kf3	Qxb3ch
42. Ke2	Qxc4ch
43. Kd2	Rf7
44. f3	a6
45. Ke3	Qc3ch
46. Ke2	Qxf3ch
47. Kd2	c4
48. Qe5	Rd7
49. Kc2	Qxd5
50. Qe1	Qg2ch
51. Kc3	Qxh2
52. Qe4ch	

"That MYCHESS could have mated Euwe a number of times previously, even Max agrees. The only reason Euwe continued was to see how it would win this easy game. But once White started checking, Black found NO WAY of getting out of check, and after a great number of checks, a draw was agreed upon! Let's claim it as a Moral win for MYCHESS over Max Euwe."

Another Sparkle from BORIS 2.5

Max Euwe was not the only grandmaster to lose to a microcomputer. Larry Christiansen suffered the same fate in another encounter reported by George Koltanowski. "Grandmaster Larry Christiansen was playing 28 opponents," comments George, "in his exhibition in Schenectady, New York on April 3, 1980. All went well for the first couple of hours. He was mowing them down, with 12 wins, and good prospects of winning most of the games still left to play. Something unfortunate happened to him then. A microcomputer, BORIS 2.5, mated him! And he had the darned machine beat. Larry ended up with 4 losses.

CHRISTIANSEN White	BORIS 2.5 Black
1. c4	e5
2. Nc3	Nf6
3. Nf3	Nc6
4. g3 (a)	Bb4
5. Bg2	0-0
6. 0-0	d5 (b)
7. cxd5	Nxd5
8. Nxe5	Nxc3
9. bxc3	Nxe5
10. cxb4	Qd4(c)
11. Rb1	Rd8
12. Bb2	Qd6
13. d3	c6
14. Qc2	Bg4
15. h3	Be6
16. Qc3	f6 (d)
17. f4	Ng6
18. Kh2 (e)	Bxa2
19. Ra1	Bd5

20. e4	Bf7	24. Kh1	b5	28. d5	Rf8 (f)
21. d4	Qd7	25. g5	Bc4	29. Bf3	cxd5 (g)
22. f5	Ne7	26. gxf6	Bxf1	30. Rg1ch	Ng6 (h)
23. g4	a6	27. Rxf1	gxf6	31. fxg6 (i)	Qxh3 mate.

Koltanowski says:

(a) 4.d4 is also played here.

(b) Not afeared to open up the game. More solid is 6 . . .Bxc3;7.bxc3,d6.

(c) Maybe the Grandmaster will not notice that his rook is en-prise. Immediately Qd6 is much better — but how does one convince a microcomputer that a Queen is a very valuable piece and should not wander too far afield — loosely!

(d) Protects and defends! (Against a possible mate on g7.)

(e) Why not 18.f5 winning a piece?

(f) Good advice is difficult to give now. After 28 . . .Qd6; then 29.Qxf6, Qxf6; 30.Bxf6,Re8; 31.d6.

(g) But here 29 . . .Qd6 was needed and then after 30.Rg1ch,Kh8;

(h) Forced. If 30 . . .Kf7; 31.Bh5ch or 30 . . .Kh8; 31.Qxf6ch, Rxf6; 32.Bxf6 mate.

(i) Even Grandmasters blunder . . .With 31.Bh5,Rac8; 32.Qd3, White wins the Knight and should win.

More on Chess and Computers

A New Book Review
BY FLOYD R. KIRK

Just how well **do** computer chess programs play? David Levy (the principal author) and Monroe Newborn have given us one set of answers in their new book, **More Chess and Computers,** a sequel to (but independent of) Levy's earlier **Chess and Computers.** Twenty-one annotated games, extending from 1975 to 1978, are presented in the text; the appendix contains another 55 games (un-annotated) from 1977.

In Chapter 1, Levy describes his now-famous bet and includes every chess game he played to win it. His first challenge did not come until 1977; he won against Chess 4.5 and later against Russian champ Kaissa. The next year, he beat the MacHack-Cheops program. The final challenge came from the Chess 4.7 program. Levy ended that match with three wins, a draw and a loss.

Chapter 2 shows some of the best of the recent computer chess games, including an impressive game between Chaos and Chess 4.4 (on this game, the annotations are similar to those of Chess 4.7 as detailed above). Having shown what computers can do, Levy then discusses their main failing — **a lack of theoretical chess knowledge.** Most computers see everything that can occur in the first six plies or so but are blind to most concepts which require long-range planning, such as zugzwang, minority attacks, bishops-and-knight mates, weak squares and color complexes.

As documented in Chapter 3, computers do much better at blitz chess. Such games, played at the rate of only seconds per move, do not allow long-range planning, so the computers' tactical abilities often predominate. Levy presents five wins by Chess 4.6, one against an International Grandmaster and four against International Masters. Levy regards those games as typical, rating Chess 4.6 at blitz chess as a strong International Master. At tournament chess, on the other hand, he would probably rate it as only a strong Class A player.

Chapter 4 and Appendix B contain all the games from the 1977 U.S. Computer Chess Championship and the 1977 World Computer Chess Championship. Levy presents one game from each in the text, with notes; the rest are found, without notes, in Appendix B.

Professor Monty Newborn wrote Chapter 5 on microcomputers and chess. He presents some of the history behind microcomptuers and explains some of their limitations. He also concludes that, with 16-bit microprocessors and high-level languages, the microcomputers will soon be playing good games of chess. He presents one game with Chess Challenger 10 (the Chess Challenger 7 did not then exist) and one with Sargon II (the Sargon 2.5 module did not exist then, either). Newborn concludes by describing two magazines: *Personal Computing,* which "provides the most extensive coverage" of computer chess, and *Byte*.

Overall, I liked **More Chess and Computers.** It is an inexpensive collection of some of the best computer chess games played through 1978; I especially enjoyed the six games which listed the computers' assessments and predictions. The book does not provide any computer chess programs nor does it give any concrete ideas for such construction. Unlike the complexity of other computer chess books, no mathematical background is required to read this one. It does provide games and commentary of interest to any chess player who plays or is thinking about playing against a computer. The chess annotation is at the level of a social player. This book, with most of its emphasis on Levy's bet and the 1977 tournaments, continues with what promises to be an even longer series of historical updates on computer chess.

"More Chess and Computers: The Microcomputer Revolution: The Challenge Match." © 1980. Authors: David Levy and Monroe Newborn. Computer Science Press, Inc. 9125 Fall River Lane. Potomac, MD 20854. 117 pages. Paperback. $12.95

Rating Boris 2.5 – Part II

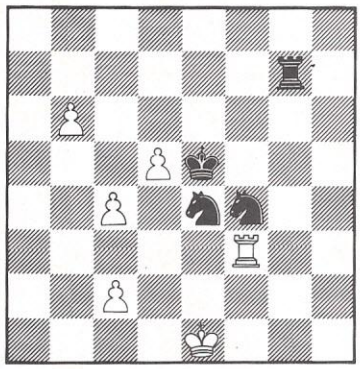
BY DAVE WELSH

(In the preceding part of this article, Engineer Welsh discussed BORIS 2.5's System, Program, Rating, Style of Playing and Program in Action. He continues here with more of Boris' playing ability and with some conclusions.)

BORIS 2.5 vs. Sargon II

A friend arranged to play his Apple II version of Sargon II against the "stand-alone" BORIS 2.5. Both have the same 7 levels of play, but BORIS 2.5 can predict a move and think on the opponent's time, and together with its faster clock speed, this allows BORIS 2.5 to play much faster. Due to the similarity between the programs, BORIS 2.5 anticipated most of Sargon II's moves and had an answer ready. Two games were played at Level 3, BORIS 2.5 winning both.

Next month Dave Welsh describes and analyzes his one-on-one contest against BORIS 2.5 at Level 4.



After Black's 36th move, White is helpless

White: SARGON II
Black: BORIS 2.5
(King's Gambit)

1. e4	e5
2. f4	exf4
3. Nf3	Bb4? (A)
4. Nc3	Nc6
5. d4	Bxc3+
6. bxc3	g5
7. d5? (B)	Nb8
8. h4	g4
9. Qd4	Qf6
10. Qe5+ (C)	Qxe5
11. Nxe5	d6
12. Nc4	f5!
13. Bxf4	fxe4
14. Bxd6 (D)	cxd6
15. Nxd6+	Ke7 (D)
16. Nxc8+	Kd7
17. Nxa7	Rxa7
18. Bb5+	Kd6
19. c4	e3? (F)
20. 0-0	h5
21. Rf8 (G)	Nd7
22. Re8	Nc5
23. a4	e2
24. Ra3? (H)	Ne7
25. Kf2	Rxe8
26. Bxe8	Ra8
27. Bxh5	Nf5 (J)
28. Kxe2	Nxh4
29. a5	Nxg2
30. Kf2	Nf4
31. Bxg4	b6!
32. Rf3? (K)	Ke5
33. axb6	Ne4+
34. Kg1	Rg8 (L)
35. Kf1	Rxg4
36. Ke1	Rg7 (M)

In this next game the computers change sides:

White: BORIS 2.5
Black: SARGON II
(English)

1. c4	c5
2. Nc3	Nf6
3. g3	Nc6
4. Bg2	e5
5. Bxc6? (A)	dxc6
6. Nf3	Bd6

BORIS 2.5 as Black

A. Sargon thinks the only possible reply is Nc3, ignoring 4 c3 and 5 d4+

B. Better is Bc4. This weakens a strong center.

C. e5 was obvious and strong.

D. This move gets three Pawns for the Bishops — bad trade.

E. Preventing Nf7

F. Sargon expects castling, ignoring Ke2.

G. Looks strong. But Black can cope.

H. Necessary was Rxe2 keeping both Rooks on.

J. Black's Knights now dominate.

K. The best chance was Bf3, hoping for . . .Rxa5. Then 33. Rxa5 with good drawing chances.

L. The point!

M. White is helpless now. (He's mated on Black's 55th move.)

BORIS 2.5 as White

A. Again BORIS 2.5 seizes the first opportunity to exchange Bishop for Knight.

B. Since its aberration on move 5, BORIS 2.5 has played very well and has some initiative.

C. Better is . . .Nd7 followed by . . .f6

D. Better was b3.

E. Black should liquidate the weak c-Pawn by . . .c4.

F. SARGON II cooperates in the weakening of his white squares. A better plan for him would have been . . .f6.

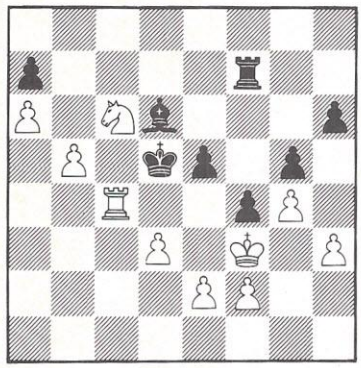
G. Both sister programs overlook . . .Qh6, winning a Pawn.

H. Bd6 was necessary.

I. BORIS 2.5 finds an elegant forced win of two Pawns.

J. Or . . .Qxc5. 32. Rxc5 Bxe4 33. Nxe4 Kg7 34. Rc4.

K. SARGON II is now helpless and can only mark time. In the course of the game's ending, it was evident that BORIS 2.5 has been programmed to avoid draws by repetition, when that pattern appeared at the 45th move. Future versions of these programs could be enhanced by a resignation function in hopeless positions. This game ended (after 66 moves) with BORIS 2.5 holding two Queens (from promoted Pawns) while SARGON had none.



After The 41st move SARGON II is again helpless.

7. d3	Be6	9. Na4	Qc7	29. Ra5	f6? (H)
8. 0-0	Qb6	10. Bg5! (B)	b5?! (C)	30. Rxc5! (I)	Rxc5
		11. Bxf6	gxf6	31. Qxc5	fxg5 (J)
		12. cxb5	cxb5	32. Qxe7	Rxe7
		13. Nc3	Rb8	33. Nf6+	Kf7
		14. a4!	b4	34. Nxd5	Rd7
		15. Ne4	Qe7	35. Nxb4	Bd6
		16. Qc2	0-0	36. Nc6	Ke6
		17. Rfc1	Rfc8	37. b4	Kd5
		18. a5?! (D)	f5?!	38. b5	Rf7
		19. Ng5	Bd5	39. Rc4	Bf8
		20. Qd2	f4 (F)	40. Kg2	Bd6
		21. g4	Qd7	41. Kf3	h6 (K)
		22. h3	Qc6		
		23. Qc2	Rb7		
		24. b3	Bb8		
		25. Nd2?	Qd6? (G)		
		26. Nde4	Qe7		
		27. Ra2	Rd7		
		28. a6	Rc6		

Whaland's "What Chess Programs Don't Do" will continue next month. Computer Games will also re-appear next month.

Classifieds

Rates for advertising in this section: $1 per word. Minimum 15 words. Allow two months for appearance (usual publication lag). Announcement of human tournaments that are open to computers published without charge. Send all submissions for this section to COMPUTER CHESS CLASSIFIED DEPARTMENT.

BACKGAMMON CASSETTE:

The GAMMON CHALLENGER gives all backgammon players a battle. Has 3 levels of play. Switches sides, tests problems and stores positions in memory. Available on TRS-80 cassette for either Level 1 or Level 2. $14.95 each. Computer Cablevision, 2617 42nd Street, NW, Washington, DC 20007.

ENTER A GOMOKU TOURNAMENT

Do you have a GOMOKU program? Would you like to write one? Those with own programs can enter an International GOMOKU Tournament and, if #1, can take on the European champ. For more information on the GOMOKU Tournament itself, write to Dr. Shein Wang, Institute of Computer Science, U. of Guelph, Guelph, Ontario, N1E 1C8.

NEW! SARGON/BORIS 2.5

Modular Game System featuring the world's strongest playing microcomputer chess program! (Plays above 1500 level in tournament time.) We have SARGON 2.5 plus the Response Board in stock and available for immediate delivery! Write or phone, 24 hours a day for free discount price list, brochure, and an in-depth technical report. Palmer, McBride and Kincaid Associates, PO Box 598, East Brunswick, NJ 08816. Tel. (201) 246-7680.

JOIN:

ICCA (International Computer Chess Association.) $10 annual membership includes the ICCA NEWSLETTER with computer-chess news from all over the world. (Back issues, $2 for set of three.) Send U.S. check or international money order to ICCA, c/o Ken Thompson, Room 2C423, Bell Telephone Labs, Murray Hill, NJ 07974, USA. Editorial material for the ICCA Newsletter should be sent to: B. Mittman, Editor, ICCA NEWSLETTER, Vogelback Computing Center, Northwestern University, Evanston, IL 60201, USA.

COMPUTER-CHESS BOOK

Learn the psychological secrets of becoming a strong chess player! "How to Beat Most People and Computers at Chess." (Secrets of a Grandpatzer.) By Kenneth Mark Colby, Professor of Psychiatry and Computer Science, UCLA. Hardcover. 153 diagrams. 141 pages. Figurine Notation. For personal autographed copy, send $20 (check or money order) to MALIBU CHESS PRESS, 25307 Malibu Road, Malibu, CA 90265. (CA residents please add 6%.)

CHECKERS FOR APPLE II

Checkers is one of four games on Disk A4 in a series of Apple II computer software written by John H. Barnes. Requires: Apple II computer – 32K RAM, One Drive – Disk II, Applesoft II in ROM – firmware card. Checkers gives average players a chance to win sometimes – the outcome is not predictable. Disk A4 is entitled "Graphics Games" and the package utilizes both high and low resolution graphics. Checkers, Tic-Tac-Toe, and Bowling use Apple's low resolution color graphics. The Derby is run in high resolution graphics. All four games come on Disk A4. It sells for $19.95. Williamsville Publishing Co., P.O. Box 250, Fredonia, New York 14063.

COMPUTER CHESS:

Select moves with programmed questionnaire (Boris). Defense, development, tactics and strategy covered. Manual, 69 pages, $4.95. R. Belanger, 35 W. Charlotte Avenue, Cincinnati, OH 45215.

THE JOY OF CHECKERS

Explore, enjoy checkers! Send for rules, pointers, sample magazine, membership benefits. Our bet: $5,000 on World Champion Tinsely against any computer! American Checker Federation, 3475 Belmont Ave., Baton Rouge, LA 70808.

BRIDGE PROGRAM

For those who would like to add my standardized bridge dealing sequence to their Duisman programs: If they will send me a check for $3 (to cover incidental costs) I shall be glad to send the code. Thomas A. Throop, 8804 Chalon Drive, Bethesda, MD 20034.

GOMOKU FOR MICROS

Play an intelligent game at university level! Five Stones Software's GOMOKU program is available for North Star or CP/M in 5-1/4" single or double density disks or in 8" single density disks. Also available in cassette for TRS-80, Level II, 32K. $29.95, any choice. Visa or Mastercharge accepted. Five Stones Software, PO Box 1369, Station B, Ottawa, Canada. K1P 5R4.

OTHELLO FOR TRS-80

Intelligent OTHELLO opponent for both novice and expert. Five levels of play, neat graphical display, move selection in 30 seconds or less. TRS-80 tape ($12) requires Level II and 16K and has REAL-TIME Lunar Lander program on flip side. Apple tape ($16) requires integer BASIC and 24K and has OTHELLO for Disk system on flip side. Peter Frey, 2407 Prospect Avenue, Evanston, IL 60201.

More Bridge on the TRS-80, TI 99/4

THOMAS A. THROOP

This month I'd like to present a few more examples of deals from the two bridge products for personal home computers I am developing with two members of the Dallas Aces bridge team. Both products are expected to be on the market around Christmas.

I'm developing the TRS-80 product with Bob Hamman of the Dallas Aces. I'm developing the Texas Instruments 99/4 product with Bobby Wolff of the Dallas Aces. Wolff is Bob Hamman's usual bridge partner in major tournament events.

Let's first look at a few deals from the new TRS-80 product. This product is a playing program, with two principal options. First, for random deals, as declarer you play the North and South cards at a contract the computer suggests or one of your choice, while the computer defends with the East and West cards. You may play a deal seeing only the North-South cards or seeing all four hands.

You may also enter into the TRS-80 a particular deal of special interest to you. Then, as before, you will play the North-South cards while the computer defends with the East-West cards.

The second principal option is to play one of a number of selected deals in an instructional mode. At each turn to play one of the N-S cards you will be given three chances to choose the best or recommended card to play.

The first deal to discuss is deal number 306 in the random deal mode. The contract suggested by the computer program is 4 spades, which is quite proper. Let's see how the computer defends while we play the N-S cards. The defensive play is that as of the time of writing this column:

NORTH
(Dummy)
♠ KJT3
♥ Q876
♦ T8
♣ AKJ

SOUTH
(Declarer)
♠ Q542
♥ 5
♦ AJ4
♣ QT765

The TRS-80, as West, opens the 2 of hearts against your 4 space contract. You play the 6 from dummy, and East plays the ten, winning the trick. East attempts to cash the ace of hearts, which you ruff with the 2 of spades.

The proper plan on this hand is to reserve the dummy by using the trumps in dummy to draw the enemy trumps while using some of your trumps to ruff some of dummy's losing cards. You should be able to make 5.

At trick 3 you can lead one round of trump. I led the queen of spades from my hand, preserving the high trumps in dummy for drawing the enemy trumps. On the spade queen West played the 8, I played the 3 from dummy, and East took the queen with his ace.

The computer as East made a shift to a diamond, leading the 5. You win with the ace and ruff one more heart in your hand before drawing all the outstanding trumps. The best play is to enter the dummy by leading the 4 of spades, on which West shows out, discarding the 3 of diamonds. I played the ten from dummy, and East played the 6.

Next, I ruffed a second heart with my last trump, both East and West following. Now you have the rest of the tricks, assuming that East has at least one club. You play a low club to dummy's ace or king, cash the king and jack of spades to draw East's trumps, cash the king or ace of clubs, and lead the jack of clubs from dummy, overtaking this with your queen. You win the last two tricks with your ten and 7 of clubs, making 5 for an overtrick. The deal:

NORTH
(Dummy)
♠KJT3
♥ Q876
♦ T8
♣AKJ

COMPUTER
WEST
♠8
♥ K9432
♦ Q7632
♣ 83

COMPUTER
EAST
♠A976
♥ AJT
♦ K95
♣942

SOUTH
(Declarer)
♠Q542
♥ 5
♦ AJ4
♣ QT765

	W	N	E	S
Trick 1	2H	6H	TH	5H
2	3H	7H	AH	2S
3	8S	3S	AS	QS
4	2D	8D	5D	AD
5	3D	TS	6S	4S
6	4H	8H	JH	5S
7	3C	KC	2C	5C
8	6D	KS	7S	4D
9	9H	JS	9S	JD
10	8C	AC	4C	6C
11	7D	JC	9C	QC
12	QD	TD	9D	TC
13	KH	QH	KD	7C

Contract: 4 spades
Tricks N-S: 11 Tricks E-W: 2

The next deal to discuss is deal number 447. The contract suggested by the TRS-80 program is 3 notrump, which is quite reasonable. Your cards and those of the dummy are as follows:

NORTH
(Dummy)
♠75
♥ AKJ83
♦ KQ64
♣ K4

SOUTH
(Declarer)
♠AQJ64
♥ 96
♦ AJ8
♣ Q83

The TRS-80 program, as West, opens the 7 of clubs against your 3 notrump contract. You play the king from dummy, and East plays the deuce. You now have eight tricks off the top. Which suit should you attack for your ninth trick? Spades. If the spade finesse loses, it will put West on lead. If the heart finesse loses, it will put East on lead, who is the player you do not want on lead through your Q8 of clubs. If the spade finesse should lose, in the process you have set up one more spade trick, assuring you of the contract.

Therefore, at trick 2, the proper play is a small spade from dummy, East plays the deuce, and you finesse with the queen, which wins the trick. Now you enter dummy with the king of hearts to repeat the spade finesse. On 7 of spades from dummy East plays the 3, and you finesse with the jack, which wins the trick.

Now, when you cash the ace of spades, East's king falls, setting up two more spade tricks in your hand. You now have seven of the last eight tricks by cashing these two spades, four diamonds, and the ace of hearts. East, discarding carefully, wins the last trick with the queen of hearts, the only trick for the defense. Here's the deal:

NORTH (Dummy)
♠75
♥ AKJ83
♦ KQ64
♣ K4

COMPUTER WEST
♠ T98
♥ T72
♦ 972
♣ AT97

COMPUTER EAST
♠ K32
♥ Q54
♦ T53
♣ J652

SOUTH (Declarer)
♠ AQJ64
♥ 96
♦ AJ8
♣ Q83

	W	N	E	S
Trick 1	7C	KC	2C	3C
2	8S	5S	2S	QS
3	2H	KH	4H	6H
4	9S	7S	3S	JS
5	TS	4C	KS	AS
6	2D	3H	5C	6S
7	7D	8H	6C	4S
8	9D	4D	3D	AD
9	9C	6D	5D	JD
10	TC	QD	TD	8D
11	7H	KD	JC	9H
12	TH	AH	5H	8C
13	AC	JH	QH	QC

Contract: 3 notrump
Tricks N-S: 12 Tricks E-W: 1

As mentioned earlier, the second principal option is one in which you play one of a number of selected deals in an instructional mode. Each time it is your turn to play one of the N-S cards you will be given three chances to select the best or recommended play. If you do not come up with this play in three choices, the computer program will tell you what this play is, and then you continue with the play of the deal.

Each of the deals in the instructional mode illustrates one or more techniques of good bridge play. These techniques include hold-up plays, safety plays, finessing into the safe hand, reversing the dummy, end plays, elimination plays and trump coups.

Here is a deal very similar to one of the deals in the instructional set, although this particular one was not included in the final set:

NORTH (Dummy)
♠ Q3
♥ A6
♦ AKT873
♣ 873

SOUTH (Declarer)
♠ K75
♥ KT53
♦ Q9
♣ AQ62

Your contract is 3 notrump. West leads the 6 of spades, you play the queen from dummy, East plays the 9, and you play the 5 from your hand. Since West presumably has the ace of spades, you must avoid letting East gain the lead and leading a spade through your king. This is a similar situation to that in deal 447 in the random play mode, which we just discussed.

I am sure that you have spotted the correct line of play. At trick 2 you simply lead a low diamond from the dummy and finesse your 9 of diamonds. If it should lose to West, your spade king is protected. You will win at least 1 spade, 2 hearts, 5 diamonds, and 1 club. As it turns out, the finesse of the diamond 9 wins, and you will make at least four notrump. Here is the deal:

NORTH (Dummy)
♠ Q3
♥ A6
♦ AKT873
♣ 873

COMPUTER WEST
♠ AT862
♥ J972
♦ 4
♣ K95

COMPUTER EAST
♠ J94
♥ Q84
♦ J652
♣ JT4

SOUTH
♠ K75
♥ KT53
♦ Q9
♣ AQ62

The TI 99/4 is a bidding program. As South, you select the bids with your cards, while the computer provides the bids for your partner, North, and your East-West opponents. Each time it is your turn to bid you have three chances to come up with the best or recommended bid. If you do not select this bid in three choices, the computer program will tell you what is the preferred bid, as well as providing you with an explanation of this bid, bids by your partner, and bids by East-West. Also, explanations of certain bidding conventions may be displayed at your request.

Let's look at a hand expected to be in the final product and see how you would bid. This hand is included in "Game Bidding". Your cards are:

♠ KJ7
♥ AJT83
♦ KJ
♣ J63

Your partner opens the bidding with 1 diamond. You respond with 1 heart. Partner rebids 1 spade. Now the problem. You have a game, but where? 4 spades, 4 hearts, 5 diamonds, or 3 notrump may be the right final contract, depending on partner's holding. How do you proceed? The correct bid is given in the TI product.

Programming Made Easy

BY KEN MAZUR

Logical Machine Corporation of California believes it has the answer to the question, "Can I use a computer strictly as a tool in my business without having to know 'computerese' or programming?" The firm's answer is a line of computer systems designed with the businessperson in mind.

Anyone who investigates computerizing a business quickly learns at least one thing: software is going to be more of a problem than hardware. Many small computers will perform adequately under business requirements; the major hangup in converting from a manual record keeping system to computers is obtaining and maintaining software (programs) that reflect the business operations of your particular firm.

You have a number of options when trying to obtain business applications software (bookkeeping, billing, record maintenance, mailing lists, etc.): buy generalized, prepackaged software and fit company procedures to the software; hire a programmer to either modify existing software or write specialized programs for your individual firm; or do the programming in-house. None of these approaches may prove satisfactory if you're a businessperson who is more interested in computerizing for efficiency than in becoming a computer enthusiast.

The first option, buying prepackaged programs, is often impractical. A business that has grown large enough to benefit from a computer already has a procedural system. Changing that system to fit the software available can range from difficult to impossible and often generates hostility among employees who find themselves having to conform to the requirements of a machine. A computer used as a tool should conform to the business, not the other way around.

Hiring a programmer to modify existing software or to write original software for your firm also has several disadvantages. First, programmers are expensive. Second, if your company has the legal right to change the coding of purchased or licensed software (and you don't always have this right), the job is generally tougher than you might be led to believe by software vendors. Coding changes can cause bugs in the programs that may or may not be discovered before they've caused problems. Most businesses have enough problems without adding to the list. Third, even a whiz of a programmer may have only the faintest inkling of business practices and procedures. Communicating your desires as to what you want in the final product can still result in a compromise between what you would really like and what the programmer perceives that you want or what he can deliver with limited knowledge of your business. You may have to install a revolving door and new carpeting by the time you get what you want and need.

Programming in-house is another tough road. Running a business successfully is a full-time job; relatively few people have the time to be successful businesspersons and expert programmers at the same time. Finding an employee with the interest and capabilities to do the job for you may be equally difficult.

Is there, then, any way an individual can use powerful small computers to a business's advantage?

Logical Machine Corporation's answer is its line of products: Adam, Tina, David and Goliath.

Though they sell a line of hardware (computers and storage devices), it is Logical's software that the company feels provides to a business a workable solution to the software dilemma.

Logical's "logical machines" (the company feels its equipment is a step above computers) are systems that allow you to computerize your own business procedures. The computer aspect and data processing capabilities of the machines are almost transparent. You don't worry about how the machine works; rather, you think more about accomplishing a specific goal by "teaching" the machine your particular business procedures than you do trying to construct a "program" that will do what you want it to do.

"Teaching" is the term Logical Machine uses instead of "programming" because an application is built through interaction between user and the machine's monitor system. The computer asks questions and you supply answers. The answers later comprise a program. All you do to create a program is supply names and numbers and tell the machines what to do by using a standard instruction set which consists of only 50 English verbs and nouns. If there is no verb that will cause a particular operation to be performed, you can define a new verb, using the standard words, which is then stored in the machines' vocabularies.

An illustration may help make the "teaching" procedure easier to understand.

Example: We wish to create a file of students that will contain a student's name and grades for science and math. We will then construct a program that allows us to enter student names and grades and a separate program to print the records we keep in the file. (User inputs are in upper case while the machine's responses are in upper and lower case.)

```
FILE called     STUDENTS
1 uses          NAME
2 and           SCIENCE GRADE
3 and           MATH GRADE
```

That's all there is to it. You don't have to worry about record lengths or whether a field is numeric or alphanumeric; the machine handles all of that internally. To create this file you simply hit the "Start" button and the file is saved to disk ready for use. You tell the machine you want to create a file, give the file a name, list the elements you want contained in the file and save it to disk.

Here's a listing of the program to enter students and grades into the created file:

```
RECAP       ENTER STUDENTS is a verb
 1 Does     CS
 2 and      DISPLAY "ENTER NAME"
 3 and      INPUT NAME
 4 and      DLF
 5 and      DISPLAY "SCIENCE GRADE"
 6 and      INPUT SCIENCE GRADE
 7 and      DLF
 8 and      DISPLAY "MATH GRADE"
 9 and      INPUT MATH GRADE
10 and      SAVE ref NAME file STUDENTS
11 and      REPEAT
```

When you tell the computer to ENTER STUDENTS, here's what happens:

The screen clears and displays "ENTER NAME" on the video. The computer waits until you enter a name, does a line feed and displays "SCIENCE GRADE"; waits until you enter the grade; does a line feed; displays "MATH GRADE"; waits for input; saves the name and two grades alphabetically by name in the file called STUDENTS and starts all over again. When you're through entering all the students, you hit the "Start" button and the machine is ready for your next project. If your program to enter students is part of a larger application (such as record keeping for a classroom), you can call up the entry portion simply by putting the verb ENTER STUDENTS (which you defined when creating the program) into the larger application.

The listing for printing the file to hard copy is:

```
        RECAP     PRINT    STUDENTS      is a verb.
        1 Does    GET           FIRST    STUDENTS
        2 and     LABEL
A
        3 and     PRINT                  NAME
        4 and     PLF
        5 and     PRINT                  SCIENCE GRADE
        6 and     PLF
        7 and     PRINT                  MATH GRADE
        8 and     PLF
        9 and     PLF
       10 and     PLF
       11 and     GET           NEXT     STUDENTS
       12 AND     GO TO         A
```

Typing PRINT STUDENTS and hitting the "Start" button gives you all the records originally entered.

Constructing the file and the separate routines to enter student names and grades and print the file took about five minutes in the very first session with a logical machine. Constructing the same sequence in Basic would take considerably longer with dimensioning arrays, alphabetizing routines, and so forth. A logical machine takes care of all the details that you have to worry about in Basic. The students could just as easily be customers; the grades, balances due.

The computers' operation vocabularies are extensive with arithmetic operations, defining operations, structure operations, testing operations (branching), library functions, data operations and file operations. These, combined with the ability to define your own operations (verbs), allow you to "program" business procedures easily as long as you know what those procedures are.

Learning the vocabulary of the machines and the functions those words perform is relatively easy, according to Victor J. Melfa, president of Victor Electronics Corp. of Southboro, MA, a New England distributor of Logical Machine systems. Within a half hour of first sitting down at the keyboard, most people can have a functioning program (application) written and working. After a 20-hour course, provided by Logical Machine Corp. at its distributors, a non-computer-oriented person is usually expert enough to program just about any application a business may have, Melfa added. Logial Machine also provides purchasers of systems with a basic stock of application programs that can be easily altered to fit the quirks of a particular business.

With a price tag of approximately $25,000 for Adam (the firm's first 10 megabyte (MByte) built-in desk unit with printer and 32K to 64K of RAM), our discussion of Logical's computers will be confined to the $15,000 Tina and $8500 David.

Tina is a four-piece unit consisting of Video Display Unit (VDU), keyboard, dual disk drives and printer. The keyboard is a standard typewriter keyboard with additional special control keys and a 15-key numeric input pad. The keyboard can be snapped into the VDU base or may be placed away from the VDU with connection by cable. The second portion of the "control console" is the VDU with a 12-inch diagonal screen that can display 24 lines of 80 characters each. Tina's brain, which is housed inside the console, is a 48K byte, 170 nanosecond, 16-bit processor. External storage consists of dual-sided, double density drives that use standard 8-inch diskettes. Capacity for the diskettes is 1.25 megabytes each for the file disk and vocabulary disk.

Tina includes a standard printer featuring upper and lower case a 7 × 9 dot matrix and bidirectional print head. Speed for the unit is 60-character lines at 67 lines per minute. Type size is 16.5 characters per inch with 8-inch maximum print width for 132 characters per line.

An alternate printer features upper and lower case, 9 × 9 dot matrix and bidirectional print head with a speed of 100 lines per minute for 60-character lines. Type size is 10 characters per inch with a 13.2-inch maximum print width to give 132 characters per line.

Tina, which may serve as a stand alone unit or as a terminal to the Goliath disk file system, has communication capability with an RS-232 serial interface built in.

Logical's latest entry to its line of products is David. David has a single disk drive carrying 1.25 MBytes of storage using dual-sided, double density disks. Memory space for the system is 64K. The VDU shows 24 lines of 80 characters each, and an upper and lower case character set with descenders. David's printer also features upper and lower case in a 7 × 7 matrix. Type size variance allows 40-, 66-, 80- and 132-column format in eight inches because character pitches of 5, 8.25, 10 and 16 can be selected within a program. Optional wider carriage and faster printers are also provided.

Logical's Goliath, a disk file storage device, has a capacity for 10 MByte, 30 MByte or 50 MByte of fixed disk storage and 10 MBytes of removable storage. The unit, which also houses the controller, may have memory ranging from 64K to 256K and capacity for up to 20 terminals. Both Tina and David computers can link to Goliath as a distributed data processing system.

While a $8500 to $15,000 price tag is a little steep for all but the most avid computer enthusiast, Melfa said the price was more than affordable for even small businesses when one takes into account the savings realized on software.

If you own a business and are thinking about getting a computer for it but aren't really interested in the machine outside of what it can do for you as a tool, you ought to check into Logical Machine Corporation's line of systems before making a purchase decision.

For more information and the address of the Logical Machine distributor nearest you, contact Logical Machine Corp., 1294 Hammerwood Ave., PO Box 60249, Sunnyvale, CA 94086; (408) 744-1290.

WHAT'S COMING UP

SOFTWARE

Forth for Apple II

Cap'n Software's Forth Ver. 1.7 includes the Forth Interest Group programming language plus extensive development aids and a 130-page tutorial manual. A telephone hot-line to system developers is also available.

Forth programs are developed interactively but run many times faster than Basic, the company said. Users write structured code to add their own operations to the language creating vocabularies optimized for their own applications. New programs are developed as modular constructs from pre-existing building blocks which may be defined by you or originally part of the language. You can also create new data types or operation types if desired. Virtual disk access provides storage for source programs, command files and data. Forth object code is extremely compact, so large programs can run in small computers.

Forth Ver. 1.7 also includes a structured macro assembler, allowing you to create machine-language subroutines if desired; these are immediately ready to run when entered, saving development time. A screen editor includes such features as automatic word wrap, delete and undelete, block move, string search and user-defined commands. Graphics and other Apple utilities are included.

The system can generate special turnkey disks which boot directly into your own application program. Developers can copy and sell their applications on these special disks with no licensing required or any other legal restrictions.

This system runs on Apple II, Apple II+ or Apple II with language card, with one or two disk drives. 48K memory is required, but over half of that is available for users' programs and data. Complete system price is $140 including all documentation. The system is available now through Apple dealers. For more information contact Cap'n Software, P.O. Box 575, San Francisco, CA 94101. *Circle No. 114*

Inventory System for TRS-80 Model II

INV-M2 is an advanced inventory control system for the TRS-80 Model II computer system. Over 7000 records can be maintained on a formatted disk exclusively used for data.

This package provides control functions to allow you to maintain an efficient inventory system with improved service at low investment costs by providing efficient order strategy and tools to measure performance. It is on-line, interactive, menu-driven and human engineered.

Order Report gives you all the inventory items at or below the safety levels and the associated order information, such as the order quantity, the vendor code and the total amount in dollars. The system also indicates priority to order. For example, order out-of-stock items first, then the high-profit items and then the high-usage items.

Performance Report provides the tools to measure the efficiency of the inventory system and the associated costs. A summary of your system includes the total inventory cost, total numbers of out-of-stock items and over-stock items and so forth.

Management Report can be produced on the screen or to the line printer. It gives you an instant summary of your inventory system, such as the total inventory cost, annual sales, cumulative numbers of times of inventory items out-of-stock or below or at safety levels, total received sales tax, total received shipping charges and so forth. It also breaks down the sales by months and projects sales volumes for the next two months based on the sales of previous months. This report is intended to give an overall summary and can be used to post numbers to the general ledger.

Other reports include data base lister and end-of-year processor, which also calculates economical order quantity (E.O.Q.) and compares to the current order quantity. In addition, a report writer is provided to allow you to specify an unlimited number of report formats on-line without any programming specified vendor.

A search command can be used to search any record on-line with a filter criterium. The searched record will be displayed and other sub-commands can be used. Global update and delete can be achieved by the report writer. Two-levels of security are used for read access only and read/write access.

This package has been field-tested and is priced at $149. A simplified version for the TRS-80 Mod-I is priced at $99. For more information contact Micro Architect, Inc., 96 Dothan St., Arlington, MA 02174; (617) 643-4713. *Circle No. 122*

Utility Software for Model I TRS-80

The Management announced a programmer utility for the TRS-80 called Fortranslator. The program is designed to aid in the literal translation of TRS-80 Disk Basic Model I programs to TRS-80 Fortran.

This machine language program will run on a 32K or larger machine with at least one disk drive. A printer is desirable.

Fortranslator converts Basic into the structured Fortran READ/WRITE/FORMAT contructs. It also translates Basic key-words and procedures such as IF/THEN/ELSE into the correct style, the company said. Fortran indentation and spacing, "C" line, "DO" loops and other conventions are produced. GOTO line numbers are created and subroutine CALL is supported. In addition, a "template" is created for Fortran specific items so you may plug-in these items after translation.

The translated program is on a diskette file that is compatible with the editor in the Fortran package. Depending on the program, at least 85% of the physical work of transfering the Basic program to Fortran is eliminated, the company said. Use of the Fortranslator means that a program can be created and debugged in Basic and then translated to the compiled Fortran for fast and efficient operation.

Fortranslator is priced at $29.95 (plus tax in Texas) and is supplied on a Model I data diskette. Instructions for use are included. A version written in MicroSoft Basic (listings only) is also available. For more information contact The Management, Box 111, Aledo, TX 76008. *Circle No. 103*

Disk Catalog Package

This program package enables you to keep track of where your software resides. It is best used for keeping track of archived files and finding out how many copies of any one program have spread throughout your collection of disks.

A directory convention is established and a directory file is set up, automatically reading the individual disk directories into the master directory file. The master file can be edited (to add descriptions of the files), merged with another, sorted into alpha-numerical order or scanned by program name, type or disk volume. Facilities are provided for dating each disk in the directory. The master file can also be scanned by the use of selective keys, such as S, ST, STA and so forth, or be printed in toto.

The package is written in Basic by Joe Kasser and incorporates an operating system in the Basic language allowing programs to be run by name (similar to CP/M). This feature is made possible by the use of the CHAIN statement whereby programs overlay memory as and when they are called in by the system.

Facilities are also available for creating invisible files, used as data or associated files that are not required to show up on a command list (help) or in the directory. For example, in an Adventure game, there is no need for all the individual files to show up in a directory listing. Invisible files are prefixed with a "*" character. The disk convention sets up the first entry as the name of the disk, the second as the date and the third as the directory. The directory is a type 3 file.

Extensive use is made of the error trapping features of the Basic as well as error detection and correction programming techniques so that the loss of data due to operator error or hardware malfunctions is minimized. Sample catalog files are provided on the disk. The instructions are simple but adequate. Price is $20. For more information contact Snow Micro Systems, Inc., P.O. Box 1704, Silver Spring, MD 20902; (301) 622-2194. *Circle No. 132*

Science Education Software

TYC Software announces its educational release The Earth Science Series containing 12 independent programs. Each is designed to teach a particular topic covered in junior high or senior high school earth science curriculum. Topics covered are: gradient, heat energy lost and gained, latitude and longitude, basic chemistry, stream erosion, water budget, seismic waves, earth history, season, meterology and percent error. A perfect supplement to a teacher's regular curriculum or for use in a general resource room, the company said.

Also included is a Lab Aid Program. This program makes the computer into an intelligent calculator, preprogrammed with 20 of the most common formulas used in lab experiments. While it helps the student with his/her calculations, it also reinforces the formulas used. In addition, there is a simple data graphing routine in which a student may create graphs of his/her lab results. None of these programs require programming knowledge.

Designed by educators to meet specific classroom needs, the programs are illustrated with numerous graphics and students are quizzed throughout to reinforce information learned. The Earth Science Series comes with a teacher/student manual which contains student objectives, worksheets, answer keys and general student user instructions. There are 12 programs on four cassettes, teacher/student manual and a vinyl storage binder. Designed for a 16K TRS-80, the program costs $59.95 plus $1.50 postage. MC/VISA accepted. For more information on the Earth Science Series contact TYC Software, 40 Stuyvesant Manor, Geneseo, NY 14454; (716) 243-3005. *Circle No. 113*

Message Software for Apple

Super Message produces advertising messages in full-page chunks on your Apple II. Each page allows statements of mixed type styles, type sizes and colors, in mixed upper and lower case.

Five type styles are available. They include regular Apple characters, double-height, double-width, both double-height and double-width, and extra-heavy double height and double-width. Six colors are available for each type style.

Centering routines automatically center the text vertically and horizontally and automatically produce a word-wrap for multi-line statements. You can chain pages of text together to make multi-page messages. Pages can be advanced manually or at user option. Multi-page messages can be stored to disk and recalled instantly.

The program requires 48K with Applesoft in ROM. Price is $50 postpaid. For more information contact Connecticut Information Systems Co., 218 Huntington Rd., Bridgeport, CT 06608; (203) 579-0472. *Circle No. 106*

Real Estate Software Package for TRS-80 and CP/M Computers

The Key Realty Management System, a software package designed for real estate operations, is now available for TRS-80 and CP/M based computers from Key Systems, Inc.

According to the company the Key Realty Management System provides the real estate executive with immediate, up-to-date information regarding the status of his/her operation. It is a combination of easy-to-use, fully documented software, created for real estate decision-makers. Since no special codes, abbreviations or 'computerese' appears to the user, no data processing expertise is necessary to get full benefits from the system, the company said.

The system is comprised of five control subsystems including Listing Control, Escrow Control, Sales Control, General Ledger and Property Management. Optional subsystems for Accounts Payable and Payroll are also available. The minimum system provides for up to 300 listings, 100 sales associates and 10 offices or sub-offices.

Listing Control, one of the primary information centers required in successful real estate operations, is provided to the realtor in a simplified form, the company said. By merely entering appropriate identifier, the realtor sees at a glance a full description of a property including dimensions of every room, floor and wall converings, special built-in equipment,

and important sales features such as location, type of property (e.g. Ocean Front).

Sales Control records the following data for each month for every salesperson and office in both volume and quantity: new listings, accepted contracts, closures, relocation commissions, commissions from listings, commissions from sales and all activities monthly for 12 consecutive months plus year-to-date, as well as this month, year-to-date, this month last year and last year to date.

Property Management is ideal for realtors engaged in professional management of an owner's property, the company said. Included are all necessary income and expense reports as well as the ability to handle up to 52 different rentals of a single property annually.

The Key Realty Management System is available from computer retail outlets across the country or directly from Key Systems, Inc. Price for software only is $2500 for the basic system. Each subsystem, as well as optional Accounts Payable and Payroll, are priced at $500 each.

For more information contact Key Systems, Inc., 16 Ocean East, Marathon, FL 33050; (305) 743-5890.
Circle No. 127

Utility Package for Apple II

The Computer Emporium has released a new utility package called Brownpak 1 for Applesoft programmers on the Apple II computer with Applesoft in ROM.

These utility programs, written by Donald Brown, include a machine language program called The Programmer's Power Tools which includes PRINT USING capabilities, packing and unpacking of data, machine-language sorting and a new input routine for disk usage.

Also included in this package are several Applesoft programs which can be used as subroutines in your own programs, the company said.

All subroutines in this package are designed for use with Applesoft programs so you can implement them into your own programs. These utilities include Automatic Diskette Menu, Disk Free utility, Hi-Res Shape utility programs and User-Defined functions from Basic. All of these routines, other than the Disk Free utility, will work from cassette.

This utility package on diskette, with instruction manual, has a suggested retail price of $39.95, which is $3.63 per utility. For more information contact The Computer Emporium, 3711 Douglas Ave., Des Moines, IA 50310; (515) 279-8861. *Circle No. 104*

Interact Machine Language Monitor

Interact computer owners have long been frustrated because they were denied access to the machine language capability of their computers. Now, advanced programmers can unleash the full power and speed of the 8080 microprocessors in their Interacts, learn 8080 machine language programming and gain a better understanding of the internal organization of their computers with Micro Video's Monitor, the company said.

With the Monitor, you can write programs that execute faster and with better color animation than can be achieved through the Basic interpreter, as well as gain access to more than 10K of RAM for programming, the company said. The program contains commands to display RAM and ROM, move segments of memory from one location to another, examine and modify the registers, substitute selectively into memory, execute programs with or without breakpoint and more. It also allows combined Microsoft Basic and machine code programming.

The Monitor program cassette with instructions is priced $19.95. For more information contact Micro Video, 204 E. Washington, Ann Arbor, MI 48104; (313) 996-0626.
Circle No. 105

Job Cost and Management Program for TI 99/4

Charles Mann & Associates released a job cost and project management system called The Project Boss for the TI 99/4 computer. The disk based system is designed to assist the project manager of construction jobs or consultants managing engineering or architectural engagements in the financial management of individual projects.

Using the system, managers can prepare job cost estimates or bids, prepare job budgets, accumulate actual job cost and estimate costs necessary for the successful completion of a project. The system allows a flexible account system for individual jobs and the preparation of reports on the budget, the costs to date, the most current estimated completion costs and the variances from the projected project costs.

The system is designed to operate with from one to three disk memory drives and any compatible printer. Use of the Disk Manager Command Module is optional. The system is available for the Apple Computer and the TRS-80 Model I computer.

The package is available from any of CMA's 650 dealers worldwide for $94.95. Additional information and dealer locations are available from Charles Mann & Associates, Micro Software Division, 7594 San Remo Trail, Yucca Valley, CA 92284; (714) 365-9718. *Circle No. 108*

Character Sets for the Pet

The math ROM character set now available from West River Electronics R&D lets the Pet computer display mathematical formulas and expressions for scientific, technical and educational use.

In the graphics mode the Pet operates normally, but in the lower-case mode, all the graphics characters have been replaced by mathematical symbols. The Pet can now display superscripts, subscripts, square roots, integrals, derivatives, sums and others.

A foreign language ROM that contains the extra characters needed for German, French, Spanish and Slavic languages is also available. Each ROM sells for $75. Although they can only be used with new model Pets, the adapter needed for most old Pets is under development. For more information on these and custom ROM development contact West River Electronics R&D, PO Box 605, Stony Brook, NY 11790.
Circle No. 131

TRS-80 Program Text Editor

A text editor is now available for standard Level II Basic from Southeastern Software.

The Southeastern Textan is a multifaceted machine language editor designed to operate with at least 16K of memory. It is a video, not line, oriented editor exclusively designed with a Basic programmer in mind, the company said.

The Textan is designed such that it reads program tapes written by Level II Basic and returns to Basic with the program fully loaded upon completion of the edit function. Its features include thirty-two command functions and twenty-six reserved word keys.

The command functions allow for top, bottom and center of screen; end of and first of line; character, word, to end of line and line delete; previous screen, word and line; next screen, word and line; search; search and replace; auto line numbering; top and end of text; line and character insert; quit insert mode; block delete; display free memory; move cursor down one line, up one line, left and right; and tape load.

The reserved word keys will automatically enter: AND, GOSUB, CHR$, DIM, ELSE, FOR, GOTO, THEN, INPUT, RETURN, KILL, LEFT$, MID$, NEXT, OPEN, PRINT, READ, RIGHT$, STRING$, TAB(, USING, VAL, DATA, REM, LEN and STR$ with a single keystroke.

Textan is priced at $40 and a manual is optionally available for $7.50 per copy. For more information contact Southeastern Software, 512 Conway Lane, Birmingham, AL 35210; (205) 956-2389. *Circle No. 128*

High Resolution Color Graphics

Cromemco has announced a new high resolution graphics software package that brings a new level of user-oriented programming convenience to the company's high resolution graphics system. The graphics system can be used to display color or black-and-white images with up to 756 x 482 point resolution on a high quality RGB monitor.

Software development for graphics systems has historically been constrained by the need to learn system-specific graphics languages, the company said. The Cromemco system, in contrast, provides a full range of powerful, human-oriented commands that operate from such common high-level languages as Basic, Fortran and Ratfor. Consequently, the Cromemco graphics software package significantly decreases programming time and thus facilitates the development of graphics applications such as process control, business statistics and charts, medical imaging and computer-aided instruction, according to the company.

The graphics software package is designed to work with Cromemco's 48KTP and 16KTP (two port) memory boards and will operate with one or two pages of two port memory. Two pages of 48K bytes of RAM are required for complete utilization of all available software options.

For those using the graphics software package, the subroutine calls provided are sufficient to fully utilize all the capabilities of the Cromemco SDI high resolution graphics interface board, the company said. These subroutines allow

you a number of powerful capabilities including: fast line generation; fast generation of regular shapes such as circles, rectangles and polygons; area fill of these shapes in a designated color at video rates; text generation and rotation; the ability to open and close windows in the page of memory being displayed; the ability to simulate motion (animation); the ability to CLIP which eliminates problems which might arise from trying to plot outside the screen area; and the ability to scale the display area of the work page.

The programmer can generate and display an image in high resolution (756 x 482 points) as well as the 16-color medium resolution (378 x 241 points) using the same system. In addition, you have the choice of plotting explicitly (i.e., specifying within a call all needed location and color information) or implicitly (i.e., specifying needed location information with regard to an implied cursor).

The software and hardware permit you to select 16 colors for the color map from a palette of 4096 colors. The contents of any color in this color map can be modified with a simple call define color command. In addition, when programming in Fortran or Assembly language, you have the option of creating color maps using the command CMAPGEN.

The color graphics package is written for both ease of use and to take full advantage of the SDI hardware, consequently, it is very efficient and extremely fast, the company said.

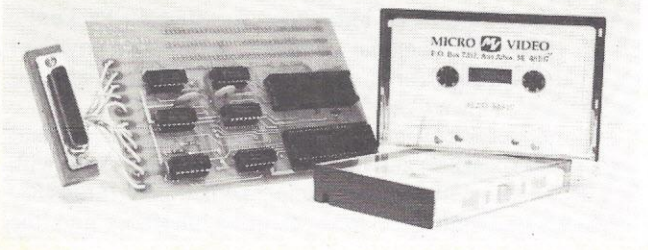
The SDI color graphics software package is available from Cromemco on either 5'' (Model SGS-S) or 8'' (Model SGS-L) diskette for $295. For additional information please contact Cromemco, Inc., 280 Bernardo Ave., Mountain View, CA 94043; (415) 964-7400. *Circle No. 125*

PERIPHERALS

RS232C Peripheral Interface

Lineprinter and communications access are now possible for Interact computer owners. Micro Video has released an RS232C peripheral interface for the Interact computer along with printer and communications software.

The interface is equipped with a dual port that has handshaking and send/receive capabilities for driving any RS232-compatible device. Installation is simple requiring no soldering. The port's design features low-power, trouble-free operations and upward compatibility with future hardware and software enhancements, the company said. All I/O parameters are completely software-selectable from Basic or machine code.

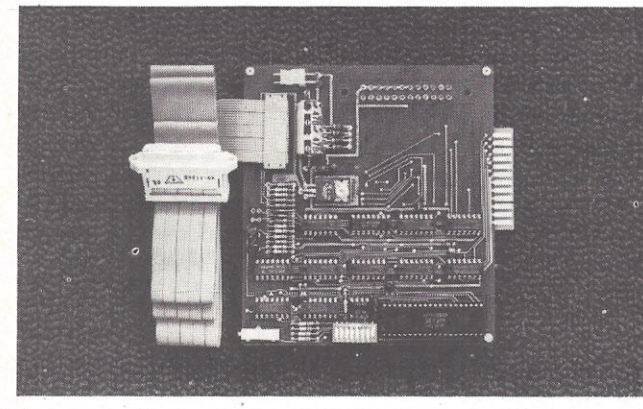


The RS232 Pack, priced at $129.95, includes Microsoft Basic with printer access commands and a Basic editor with automatic linenumbering, resequencing, string substitution, appending and other functions. Communications software is available separately. For further information contact Micro Video, 204 E. Washington, Ann Arbor, MI 48104; (313) 996-0629. *Circle No. 115*

Printer Interface for Pet/CBM

TNW Corporation announced the TNW-1000, a printer interface for the Commodore Pet/CBM. This new addition to TNW's family of IEEE-488 bus serial interfaces provides an output-only serial port to interface to current-loop as well as standard RS-232 printers and other devices.

While the TNW-1000 can be used with IEEE-488 capable computers other than the Commodore Pet/CBM, it is designed to mount right to the back of the Pet's cabinet. An edgeboard connector is provided to allow daisy chaining of other devices on the IEEE bus with the TNW-1000.



You can set the TNW-1000's baud rate over the range from 110 to 9600 bits per second, and can switch select automatic conversion from Pet to ASCII character sets for either new or old style Pet, data length and parity (8 bit words without parity or 7 bit words with even or odd parity) and IEEE bus address.

Priced at $129, the TNW-1000 is delivered fully assembled and tested, complete with power supply, 3 foot cable with female EIA (RS-232) connector and full documentation. For more information contact TNW Corp., 3351 Hancock St., San Diego, CA 92110; (714) 225-1040. *Circle No. 102*

Interfaces Give HP-85 Flexible Controller Capabilities

Three new interfaces introduced for Hewlett-Packard's HP-85 make this personal computer a flexible, powerful low-cost controller for a wide variety of instruments and applications, Hewlett-Packard said.

The interfaces are serial (RS-232-C compatible), general purpose IO (parallel) and binary coded decimal. Along with the HP-IB (Hewlett-Packard's implementation of IEEE-488-1978 standard) interface card, the new interfaces give the

HP-85 the complete set of general interfaces offered on all HP desktop computers.

With the HP-85 interfaces, the personal-computer user now can communicate with a wide variety of peripherals and instruments including low-cost serial printers and many types of measurement devices.

The serial interface card provides the HP-85 with bit-serial asynchronous data communication capability. Both RS-232-C operations and current loop operation are supported.

Features of the serial interface card include programmable line characteristics which let you change baud rate, parity, bits per character and stop bits without changing physical switch settings. Other features include true full-duplex with I/O buffers to prevent loss of incoming data while transmitting, and full 20 mA current loop, so the HP-85 can interface with almost any 20 mA current loop device such as teletype-compatible terminals and peripherals.

The serial card also gives the HP-85 a low-cost printer solution: Users who need a peripheral printer in addition to the HP-85's integrated thermal printer can now use a serial printer, some of which cost as little as $1000.

Communication through a modem over phone lines is also possible with the serial interface card.

A general purpose IO (GP-IO) interface card provides bit-parallel byte- and word-oriented interfacing, useful where a simple, wide data path interface is important. Two low-power bidirectional ports and two high-current output-only ports on the card make it possible to attach up to four devices per card.

Users can configure the card as four separate 8-bit ports, two 16-bit ports or two 8-bit and one 16-bit ports.

Common uses of this card include interfacing to a variety of peripherals such as printers, paper tape readers and punchers and card readers.

A binary coded decimal interface card is a specialized digit-parallel interface: All data is present simultaneously on a set of 48 wires. Instruments can output up to 11 binary coded decimal (BCD) digits, more digits than on any other standard BCD interface. Enough digits should be available for almost any application.

The card can accommodate two BCD instruments. BCD interfaces are used primarily in applications where the instrument was designed for immediate display. Typical instruments that require BCD interfaces are voltmeters, counters, medical equipment and electronic scales.

The I/O ROM which provides the Basic commands to access the cards' capabilities, snaps into a ROM drawer which plugs into the HP-85. The I/O ROM was introduced earlier.

The four interface cards also plug into the back of the HP-85. In some applications, the serial interface card may be used to communicate to a serial printer using the already introduced Plotter/Printer ROM ($145) and no I/O ROM.

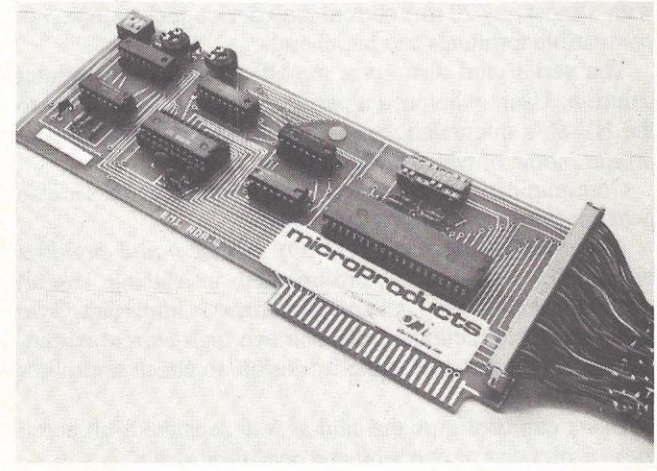
Prices are: HP-IB Interface Card, $395; Serial Interface Card, $395; General Purpose IO Interface Card, $495; Binary Coded Decimal Interface Card, $495; I/O ROM, $295; and ROM Drawer, $45. For more information contact Inquiries Manager, Hewlett-Packard Company, 1507 Page Mill Rd., Palo Alto, CA 94304; (415) 857-1501.

Circle No. 112

Data Acquisition for Apple

Microproducts announced their new Data Acquisition and Distribution board for the Apple computer. This product contains two 8 bit 15 micro-second resolution time Analog to Digital channels and two 8 bit 2 micro-second settling time Digital to Analog channels. Bipolar and unipolar reference level changes are dip switch selectable for both input and output, the company said.

This product requires no wait loops in either Basic or assembly language allowing very simple, straightforward programming. Examples are supplied in the accompanying documentation. The board will run in any Apple slot allowing multiple boards to convert the Apple to a cost effective industrial process control device.



A few of the possible applications for this board are: graphic representation with the appropriate amplifiers for EMG, EKG and EEG; laboratory monitoring and control; automatic telescopic drives in hour angle and declination with feed back; stereo music synthesis; experimenting with multiple voice speech synthesis; software development for industrial control processes; electro-acupuncture with physiological feedback; and X-Y chart recorders and digitizers.

This board with documentation and a simple 4 voice music synthesis program sells for $249.95.

For further information contact Microproducts, 30420 Via Rivera, Rancho Palos Verdes, CA 90274; (213) 541-5131.
Circle No. 110

Dot Matrix Printer

TEI Inc., announced its Model 3431 heavy-duty, 150 character/second, dot matrix printer.

The Model 3431 printer uses only heavy duty motors, printheads and other mechanical parts. The unit is powered with one of TEI's ultra-reliable constant-voltage transformer power supplies for protection from "brownouts" and line surges.

The Model 3431 printer offers up to 136-column printing using a 9 × 7 dot format to form 94 ASCII characters, including lowercase letters (with descenders), symbols, double-wide characters and so forth. It also offers true bi-

directional printing under microprocessing control. The pin-feed tractor accepts continuous forms from 1.5″ to 14″, plus multipart (to five sheets). The Model 3431 is available with standard parallel interface or optional RS-232C serial interface.

Unit price is $1695. For more information contact TEI, Inc., 5075 S. Loop E., Houston, TX 77033; (713) 738-2300. *Circle No. 111*

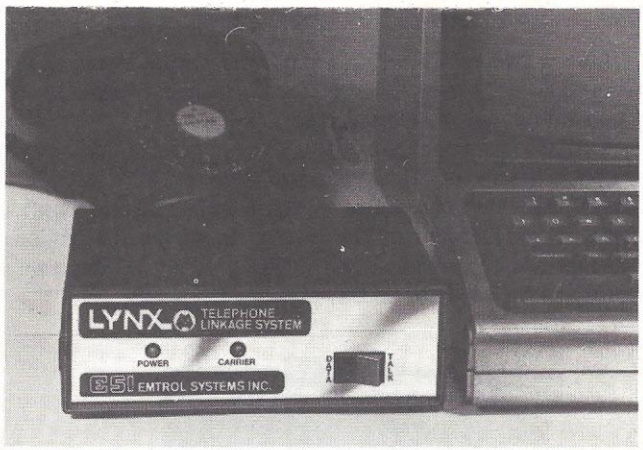
Direct-Connect Modem for TRS-80

Emtrol Systems, Inc. has introduced Lynx, a new direct-connect telephone modem designed for the Radio Shack TRS-80 microcomputer.

Lynx comprises a total telephone linkage system in one package, eliminating the need for separate expansion interface, interface board, telephone coupler and communications software, the company said.

Lynx connects directly with the TRS-80 keyboard and the telephone line; no acoustic coupler is used. It includes originate and answer capability, and is programmable for word length, parity, number of stop bits and full or half duplex.

Minimum hardware requirements for Lynx are a TRS-80 Level I or II with 4K RAM. Pending FCC registration permits direct plug-in connection with the telephone line.



During data exchange, Lynx automatically disconnects the local telephone handset, thus eliminating room noise pickup typical of acoustic couplers.

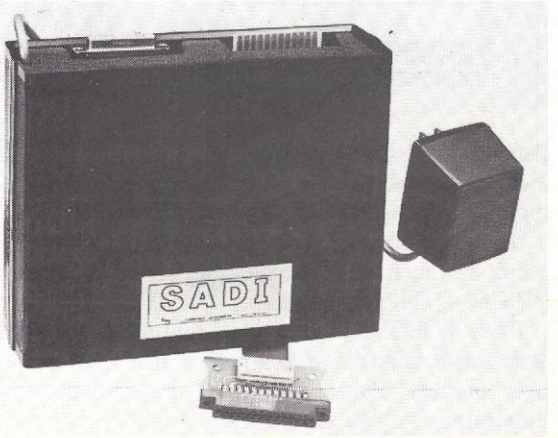
The Lynx instruction manual describes time-share access methods, such as "The Source," CBBS, Forum-80 and TRS-80-to-TRS-80 links.

Price is $239.95 including "terminal" program on cassette, instruction manual and power pack. For more information contact Emtrol Systems, Inc., 1262 Loop Rd., Lancaster, PA. *Circle No. 100*

Two-Way Interface for Pet Microcomputers

Sadi, a two-way RS-232 and parallel output interface for Commodore Pet microcomputers, is currently offered by Connecticut microComputer, Inc.

Sadi allows connection of a Pet to parallel and serial printers, CRTs, modems, acoustic couplers, hard copy terminals and other computers. Independent serial and parallel ports allow communication with both peripheral devices simultaneously or individually. Sadi also allows the transferring of programs between Pet microcomputers.



Special features for the Sadi interface include conversion to true ASCII both in and out, cursor controls and function characters specially printed, selectable reversal of upper and lower case, Pet IEEE connector for daisy chaining and addressability for use with other devices. The Sadi serial interface offers selectable baud rate from 75 to 9600, half or full duplex, 32 character buffer, X-on/X-off automatically sent and selectable carriage return delay. In addition, the parallel interface provides both data strobe and device ready in either polarity, plus Centronics printer compatibility.

Each Sadi interface comes complete with power supply, Pet IEEE cable, RS-232 connector, parallel port connector and case. Fully assembled and tested, the 110 VAC Sadi carries a suggested retail price of $295, with a 230 VAC version offered at $325. For further information contact Sadi, Connecticut microComputer, Inc., 34 Del Mar Dr., Brookfield, CT 06804; (203) 775-4595. *Circle No. 116*

TTL Based Interface

Micromatic Corporation has introduced the Micromatic 80, a TTL based interface designed to integrate the TRS-80 and many other small computers. The Micromatic 80 consists of an IBM Selectric computer printer, which is thoroughly cleaned and functionally checked due to previous

use, combined with a sleek compact interface.

The printer has a speed of 8 to 9 cps, with high quality typewriter print suitable for word processing. The Micromatic 80 simply connects to the keyboard interface port or to

the expansion interface. All code conversions and timing software are contained within the Micromatic 80 and no special software is required. The interface is warrantied for 90 days.

The Micromatic 80 is priced at $795, and can be ordered by mail or telephone from The Micromatic Corporation, 5747 West 85th St., Indianapolis, IN 46278;(317) 299-8614. *Circle No. 117*

Floppy Disk Drives for TRS-80 Model II

Parasitic Engineering announced that its Maxi-Disk 8" floppy disk drives are compatible with the TRS-80 Model II. When used with the Model II, Maxi-Disk drives are functionally identical to Radio Shack expansion drives and no software or hardware changes are needed, the company said.

Maxi-Disk drives have activity lights, a feature which helps protect against errors by informing you when it is safe to remove diskettes.

Each Maxi-Disk drive is completely self-contained in its own cabinet. Additional drives are simply plugged in. Also, a drive can be removed for service without disturbing any other drives on the system.

The compact grey and black cabinet saves room. Three Maxi-Disk drives take up only one-half the space of Radio Shack's three drive box, the company said. And, of course, one or two Maxi-Disk drives use even less space.

All Maxi-Disk drives are compatible with the Model II. Current Maxi-Disk system owners need only purchase a cable from Parasitic Engineering in order to use their drives on the Model II. Maxi-Disk drives for the Model II cost $845. A three drive cable (one needed) is $60. Delivery is approximately 60 days ARO. For more information contact Parasitic Engineering Inc., 1101 Ninth Ave., Oakland, CA 94606; (415) 839-2636. *Circle No. 107*

SYSTEMS

Hand-Held Computer Terminal Communicates Over Phone Lines

Travelling executives can now enter and receive computer data over standard telephone lines using a hand-held, portable data terminal. Based on the Nixdorf LK-3000 Personal Computer, the system uses interchangeable modules to accomplish a variety of tasks.

When used with the newest electronic module and appropriate user software, this unit can communicate with any business or private computer by means of a standard RS-232 telecommunications interface. Simply by placing the telephone receiver on an acoustic coupler and plugging in the LK-3000 with communication module, you can key-in data, review it on a 16-character scrolling LED display and then transmit to the computer. Similarly, you can access the computer to receive stored data, Nixdorf said.

Providing all the portability of a pocket calculator with the total capability of a home-base computer, this new system has applications for travelling executives, salesmen, field service personnel and others who require access to critical, day-to-day decision-making information, the company said.

Other modules available for the basic LK-3000 Personal Computer transform the unit into a language translator, filing system, calculator or programmable electronic note pad.

The LK-3000 with communications module (LK-2010) and acoustic coupler sells for $525. Interchangeable modules for other functions are priced from $35 to $100. The terminal is RS-232 compatible and operates at a baud rate of 110 or 300 bits per second, asynchronous, half or full duplex.

The terminal display is a 16-segment alphanumeric LED with integrated MOS-LSI display controller. It uses an 8-bit word length including parity, three stop bits and an 80 character display buffer.

For further information contact Phil DeNapoli, National Sales Manager, Nixdorf Computer Personal Systems, Inc., 168 Middlesex Turnpike, Burlington, MA 01803; (617) 273-0480 or (800) 225-1992.
Circle No. 119

Desktop Pascal Graphics Computer System

Demand for lower-cost graphics capabilities has resulted in a new product from Integrated Research and Information Systems (IRISystems) Corporation, the Ensemble I20GX. The new product is a self-contained computer system, packaged in a configuration that might easily be mistaken for a terminal, the company said.

Standard versions are based upon the Western Digital Pascal MicroEngine, high resolution graphics, a 15″ monitor, 12-slot S-100 standard motherboard, detachable keyboard, double density, double-sided dual floppy disk subsystem, Z80 alternate on-board MPU, memory parity, 128KB dynamic RAM, UCSD Pascal, CP/M, constant voltage transformer and printer port for graphics hard copy output. Single quantity price is $9796.

The Ensemble I20GX can present a 768 × 480 pixel format in 8 × 16 pixel characters; some of these can be user-defined.

The Ensemble I20GX in its standard configuration runs under UCSD Pascal and CP/M. Alternate configurations run only under CP/M, do not use the MicroEngine, but offer a Pascal compiler which generates floating point instructions for the AMD9511 chip. A low-cost version of the Ensemble I20GX uses the Z80 MPU, a 9″ black and white monitor and one 5.25″ floppy disk drive. Packaging includes a 12-slot motherboard, CVT power supply and built-in expansion for two more 5.25″ drives. It costs $3464.

MicroPro's WordStar and other CP/M-supported software are also

available. In particular, communications protocols are supported which make the device a flexible desk-top computer/terminal, the company said. Users can send and receive disk files from host systems, as well as emulate dumb terminals. An integral FCC-registered 300 baud modem with auto-dial is available.

A flexible trade-in policy permits users to receive substantial discounts on future purchases, such as upgrades of the disk memory systems. Maintenance is available in several major cities. For more information contact Integrated Research and Information Systems, 10150 Sorrento Valley Rd., Suite 320, San Diego, CA 92121; (714) 457-3730. *Circle No. 120*

New Entry Level Business System

Microtech Business Systems introduced the System 50 — an entry level business system geared to the first time computer user or companies that have been working with an accounting or ledger card billing machine.

The System 50 features a high speed 16 bit Nova compatible minicomputer with 64K bytes of memory, industry-standard 10 Megabyte hard disk drive with removable cartridge, 24 line video display and the S-150 printer — 150 character per second, bi-directional, 132 column matrix printer. An additional video display or communications port can be added.

Included in the basic price of $14,500 is Iris, the Interactive Real Time Operating System, with business Basic and utilities. This means the System 50 will run all applications developed for the System 200, 300, 400 and other Iris-based computer systems, the company said.

A variety of applications software products are available for the System 50 including medical/dental, the Microtech Business Management System for parts distributors, CPAs, attorneys and a BOMP/MRP system for manufacturers.

Deliveries are 60 days ARO, however, since initial production is limited, the 1980 delivery schedule will be on an allocation basis. For additional information please contact Bill Gallucci, Microtech Business Systems, Inc., 3176 Pullman St., Suite 108, Costa Mesa, CA 92626; (714) 557-8640. *Circle No. 121*

New System Offers Upgradability

Vector Graphic Inc. announced a new highly intelligent entry level computer system. The system, being referred to as VIP (Vector Intelligent Partner) is specifically designed to allow the user to grow into the full range of Vector products as individual demands warrant.

According to Vector, the VIP's uniqueness is its upgradability, offering state-of-the-art word processing and list management capabilities and allowing the user to grow into Vector's other systems simply by adding disk capacity. In essence, this is an entry level system that provides the business user, as well as software developer, an investment opportunity in a system that grows with future requirements, the company said.

The VIP's upgradability derives from its use of the same core unit as in Vector's higher priced systems — the Vector 3, an integrated video console with built-in S-100 electronics. The single 315K mini-disk drive is housed separately, allowing addition of up to 3 more identical drives, or replacement by Vector's 2 Megabyte "Dualstor" diskette unit, by its 30 Megabyte "Megastor" Winchester drive, or by any other Vector drive units.

Product availability will be off the shelf. Vector dealers are currently being prepared for immediate delivery of the system. The price for the VIP is $3695. For more information contact Vector Graphic, Inc., 31364 Via Colinas, Westlake Village, CA 91362; (213) 991-2302. *Circle No. 123*

M.G. Kelly & Associates, Inc.

LASER ACTION PRINTER

Printers, Interfacings Consultations
Retail Price: $719.00
Wholesale Distributors Welcome

Do you think lasers are the thing of the future? The future is here now with a new light weight, innovative printer. LAP-32, Laser Action Printer, incorporates a laser diode instead of the conventional impact or thermal systems.

Standard Features and Programmable Options

*Switch Selectable-RS232C, Current Loop, And Parallel I/F
*Parallel I/F Compatable With Centronics, And Data Products
*Software Control Printer Select-Deselect, Graphics Mode Enable-Disable
*Variable Paper Widths From 3-1/2-14 Inches
*132 Columns Printing Bidirectionally
*132 Characters Per Second
*300-1200 Baud 7 Bit ASCII With RS232C I/F
*96 Upper-Lower Case Characters
*Combination Tractor-Roller Paper Feed

If you are interested in more information or wishing to purchase the LAP-32 Printer, please contact your nearest Sales Representative.

4744-B.N.E. 28th Street
Fort Worth, TX 76117 **(817) 534-5932**

CIRCLE 36

MINI-DISKS

Verbatim

10 for $29.00

SCOTCH or MEMOREX: 10 for $31.00

Checks and Bank Cards OK

Free Catalog

Disks, Etc.

P.O. Box 327
Center Valley, PA 18034
Tel: 215-439-1717

CIRCLE 37

EDUCATIONAL MANAGEMENT SOFTWARE FOR THE APPLE-II

DPC

IMMEDIATE DELIVERY ON SPEC. ED., IMMUNIZATION & FIXED ASSETS INVENTORY

- SPECIAL EDUCATION ADMINISTRATIVE PACKAGE
  - Student data, Inventory of Exceptionalities, Supplemental Services and Dates
  - Teacher Class Rosters
  - Reports totals & averages of days until diagnosis and placement
  - Students printed by specific catagory $275.00
- IMMUNIZATION ADMINISTRATIVE PACKAGE
  - Personal student data & dates for 6-DPT, 5-Polio, Rubiola, Rubella, and Mumps
  - Automatically calculates next shot date
  - Prints overdue immunization list by grade $275.00
- FIXED ASSET INVENTORY SYSTEM
  - Full maintenance — Addressed by tag #
  - Reports titled by operator
  - Full descriptive data retained — P.O.#, Model Serial #, Repairs, Appraisal, etc.
  - Prints single record, Selected range, or all records; also prints on cards $275.00

ALSO AVAILABLE IN MID-OCTOBER

- BUDGETARY ACCOUNTING PACKAGE $375.00
- ATTENDANCE ACCOUNTING PACKAGE $275.00
- GRADE ACCOUNTING PACKAGE
  - Report cards and Student transcript labels $300.00
- CENTRAL ACTIVITY ACCOUNTING PACKAGE $375.00

DEALER INQUIRIES INVITED

2405 San Pedro NE, Albuquerque, NM 87110
505-884-6042

CIRCLE 38

Free! Bubble Sort Program With

Catalog & Samples .....$ 4.00
(Applies to purchase)

DATQWIK (C)
Money Maker System $189.00
For TRS-80 (TM)
Adaptable to other microcomputers

Software development aids

TUBE80 – Display aid .....$ 49.00
NEWLIST Print programs
w/title & date .....$ 49.00
PASCII – Disk to LP .....$ 19.95
LPDIR – Dir on printer .....$ 19.95

Midwest Computer Services
28 Hiway 75
Clearwater, MN 55320k
Call: 1-612-558-2844
TRS-80 is a trademark of Tandy Co.

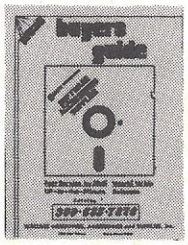
CIRCLE 39

Apple® II

SOFTWARE ACCESSORIES SUPPLIES

ULTRA-CURRENT

BUYERS GUIDE 850 ITEMS



TO ORDER
send $3.00
You receive current BUYERS GUIDE and a certificate worth $3.00 off of your first purchase.

weas **2513 N. Sheridan Rd. Peoria, Illinois 61604**

CIRCLE 40

INTERFACE EXPANSION BOARD

*For the TRS 80**

MD
MICRO-DESIGN

The Interface Expansion Board gives your computer these features:

Phone Modem
2k E-Prom Option
32 k Memory
Real-Time clock
Parallel Port
Dual Cassette Line
*Floppy Disk Controller**
On Board Supply
Silk Screen
Solder Mask
Expansion Port
Manual

MDX-1
PC Board & user manual

64.95

Add 3.00 for shipping & handling. Texas residents add 5% sales tax.
* MDX-2 W/FDC Available Nov. 80.

FREE PAMPHLET AVAILABLE
Call or write
MICRO-DESIGN
P.O. box 18054
Austin, Texas 78760
1-512-458-2937
**TRS 80 is a trademark of Tandy Corp.*

CIRCLE 41

TRS-80 & OTHER NEEDS FILLED FOR LESS

+++Compatible disk drives with power supply and case — 120 day warranty+++
* 40 track (204,800 byte/disk) use both sides, anti-crimp/power protect $329
* 8 in. drive & P.S./case $749 with P.S./case for 3 drives $929
* 80 track (204.8K byte) 90 day warranty $479
* 4-drive cable $28 ** 10 disks-5-in. @ $24-8 in. @ $36-hard case $3 & $5
*** Base 2 printer-(60 line/min. 73,80,96,120 or 132 char. line
Bi-direct dot matrix impact 96 char ASCII, 15 baud rates) $499
2K buffer, graphics, & tractor option avail.***
* Centronics 737 to $829++cable @ $25
* Selectric (word processing-typewriter & printer) $790
* Lower case foe Centronics 779/Radio Shack line printer 1-easy install $99.95
* UPS (uninterruptible power supply) prevent power drop surge or out? From $195
* Cat modem (orig/ans) $144 ++ 16K memory set (200 nano $44
* Apple, Atari, Radio Shack model 1/2 hardware/software discounted
A/R, A/P, G/L, P/R for $360 or $100 ea. (model 1) & $630 or $165 e. (model 2).
Applications interact & are complete & professional
Will run on other computer. This is a special introductory price.
* Ask for free flyer with our low prices — dealer inquiries invited Mass. residents add 5% tax-F.O.B. Tewksbury — freight extra.
M/C, Visa or check accepted. *TRS-80 is a reg. trade mark of Tandy Corp.*

OMNITEK SYSTEMS
24 Marcia Jean Dr., Dept. K, Tewksbury, MA 01876
Call 617-851-3156

CIRCLE 42

SAVE MORE THAN 20%
NORTH STAR — INTERTUBE — MICROTEK
ZENITH — RCA — COSMAC — ITHACA
THINKER TOYS — GODBOUT — SUPERBRAIN
THE SMARTEST COMPUTERS AT THE SMARTEST PRICE

HORIZON-1-32K-DOUBLE DEN ASSM & TESTED $1994
HORIZON-2-32K-DOUBLE DEN ASSM & TESTED 2299
2 NORTH STAR SOFTWARE DISKS w/HORIZON .....FREE

NORTHWORD	$ 299	MAILMANAGER	239
INFOMANAGER	369	GENERALEDGER	799
HRZ-64Q+HARDDISK	7199	HRZ-2-32K-Q ASM	2699
NS HARD DISK 18M	3939	NS PASCAL ON DISK	190
NORTH STAR BASIC	FREE	COLOR!!	PHONE
ITHACA COMPUTER	2695	ITHACA RAM 64K	845
8086 CPU 16 BIT	556	SSM Z80 CPU KIT	221
MEAS 64K RAM ASM.	599	SSM VIDEO BRD VB3	412
GODBOUT 16K STAT	285	SUPERRAM 32K	580
DISCUS/2D + CP/M	975	SUPERRAM 16K	290
2D ADD DRIVE	650	2+2 ADD DRIVE	975
TARBELL CONTROLLER	295	DISCUS/2+2	1259
INTERTUBE II	725	HARD DISK 26 Mb	3995
SUPERBRAIN	2395	SUPERBRAIN QUAD	2995
ZENITH Z-19 A&T	739	ZENITH H-11	2995
HEATH Z-89 48K	2495	MICROTEK PRINTER	675
ANADEX 9500-1	1389	ANADEX 8000	865
NEC PRINTER	2799	SECRETARY WORD PRO	77
TEXTWRITER III	112	GOFAST SPEEDBASIC	71
PDS Z-80 ASSEMBLER	89	NS BASIC COMPILER	??
EZ-80 Z-80 TUTOR	25	EZ-CODE IN ENGLISH	71
ECOSOFT ACCOUNTING	315	WORDSTAR	325
BOX OF DISKETTES	29	BASIC DEBUGGER	89

Which Computers are BEST? BROCHURE.....FREE
ORDER 2 or more COMPUTERS...BIGGER DISCOUNTS
WE WILL BEAT OUR COMPETITION'S PRICE
FACTORY ASSEMBLED & FACTORY WARRANTY

AMERICAN SQUARE COMPUTERS
KIVETT DR. * JAMESTOWN, NC 27282
919-889-4577

CIRCLE 43

WHY PAY CUSTOM PRICES???

SMALL QUANTITY
STOCK CONTINUOUS

FORMS

- CHECKS
- INVOICES
- STATEMENTS
- SPEED-O-GRAMS
- BILLS OF LADING
- PURCHASE ORDERS

IMPRINTING AVAILABLE

STOCK PRINTOUT PAPER
STOCK CONTINUOUS LABELS
Call or Write for Prices & Samples

DDF DISCOUNT DATA FORMS, INC.
407 EISENHOWER LANE SOUTH
LOMBARD, ILLINOIS 60148
(312) 629-6850

CIRCLE 44

New System Marks Entry Into Personal Computer Market

Casio, Inc. introduced the FX-9000P console, marking its entry into the personal computer market.

Casio said marketing and distribution of the product is not expected before February 1981 with estimated suggested retail price to be under $900.

One of the features of the FX-9000P personal computer is the RAM (random access memory) package with its power back-up. This allows instantaneous operation of the user system when the power is switched on. According to Casio, this feature eliminates the need for IPL from magnetic tape and improves reliability.

A video-age graphic display system makes it possible to display graphs, diagrams and tables. Casio said powerful statistical processing programs are capable of performing regression analysis, standard deviation and calculating coefficients of correlation. The FX-9000P has all functions necessary to perform scientific and technical calculations and business analyses. In addition, slot-in memory packages allow you to expand memory capacity to meet various requirements.

For more information contact Casio, Inc., 15 Gardner Rd., Fairfield, NJ 07006; (201) 575-7400. *Circle No. 126*

COMPLEMENTS

Product & Service Expansion

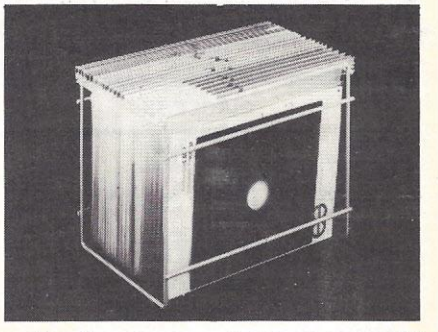
Cybermate has announced sale prices for its products and services and an expansion in the scope of its services. Its package of 41 program listings for the TRS-80 Model I Level 2 with 16K covering games, natural language, astronomy, graphics, data base management and word processing has been reduced to $9 while individual program listings have been reduced to 90¢ each and individual programs on cassette have been reduced to $2.50 each. Orders received from outside the U.S.A. must include $1.50 for postage and handling.

Cybermate's monthly "unusual programming" publication for the TRS-80 Model I Level 2, The 80 Notebook, has reduced its subscription rates to 95¢ for a sample copy, $11 per year in the U.S.A., $16 per year in Canada, and $23 per year for air mail delivery outside the U.S.A. and Canada.

Cybermate has expanded its operation to be available for business, scientific, education, operating systems and general programming assignments on a contract basis at a limited time special rate of $11 per hour of work required. All programming assignments must be for a TRS-80 Model I Level 2 with the cassette tape, minifloppy disk and sufficient memory recommended for the job at the time a quotation is given. The programming can be done in either Basic or Assembly language and will include complete system, program and user documentation. All contract programming is guaranteed to meet the specifications requested with program maintenance services available. To order or for further information on any of the above products or services, write to Cybermate, 5967 Sullivan Trail, Nazareth, PA 18064; (215) 759-6873. *Circle No. 129*

Diskette and Cassette Holders

The new Eichner diskette Desk Organizer holds ten diskettes individually in flat pressure free pockets. The



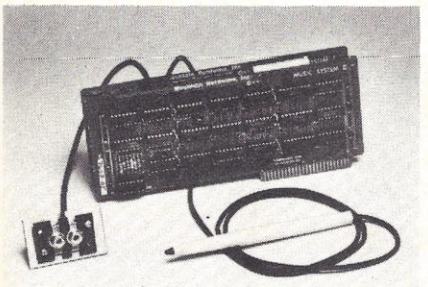
antistatic property of the raw material guarantees the safest possible storage environment for sensitive magnetic media. Each pocket has an index strip for ease of filing.

The Eichner Mini Basket, available in both letter and legal sizes, holds 20 cassette folders or 30 diskette or magnetic card folders.

Price for the Desk Organizer is $70. The letter size Mini Basket sells for $18 and the legal size for $22. For more information contact Eichner Systems, Inc., 1460 Industrial Drive, Itasca, IL 60143; (312) 773-1881. *Circle No. 130*

Music Synthesizer for Apple II

Mountain Computer Inc. has announced MusicSystem for Apple II computers. This 16 voice digital synthesizer creates sounds like real musical instruments utilizing the principle of additive synthesis. The generation of sounds is accomplished through fully programmable waveforms, envelopes, and amplitudes for each musical voice.



Provided with the hardware system is software for editing and playing of musical compositions. The Editor program permits graphical input of sheet music utilizing standard music notation. The Player program permits polyphonic performance of musical compositions. Stereo output is to users' stereo amplifier and speakers, or directly off card with stereo headphones.

For further information contact Mountain Computer Inc., 300 Harvey West Blvd., Santa Cruz, CA 95060; (408) 429-8600. *Circle No. 118*

Word Processing Report

The Small Systems Group is publishing a series of in-depth product evaluation reports; the first report entitled "Word Processing on Personal Computers," is now available.

This report introduces personal computer word processing with sections on software, hardware and applications. It continues with general descriptions of four programs: Auto Scribe, Electric Pencil, Magic Wand and Wordstar. These are compared on quality of documentation, ease of learning, editing power and formatting power. In making the comparisons, 159 features were analyzed, and the final section tabulates and discusses these in detail.

Single copies are available for $10, or $12 outside of North America, from the Small Systems Group, Box 5429, Santa Monica, CA 90405. California residents must add 6% tax. *Circle No. 133*

Supplies and Accessories Catalog

Minicomputer and small computer system end users looking for fast turnaround of quality, name brand computer supplies and accessories may now select from over 300 such items in a new catalog available from Challenge Computer Supplies, Inc.

Challenge Computer Supplies' Fall 1980 Catalog describes a full line of reputable, high performance products to meet a wide variety of computer user requirements. Among the products shown are Memorex magnetic media, Write Line cabinets and Moore Business Forms paper.

To obtain a free copy of the catalog or for additional information contact Robert D. Leigh, Challenge Computer Supplies, P.O. Box 3269, Redwood City, CA 94064; (415) 365-8105 *Circle No. 134*

Software Summary Guide

Rainbow Associates has announced the CP/M Software Summary Guide, a concise, handy summary of the major software used on most CP/M systems. Included are summaries of the CP/M operating system, Microsoft Basic, CBasic, and the CP/M utilities DESPOOL, MAC and TEX.

The CP/M summary covers the commands (DIR, ERA, REN, SAVE, TYPE, USER) and utilities (ASM, DDT, DUMP, ED, LOAD, MOVCPM, PIP, STAT, SUBMIT, SYSGEN, XSUB). Each command or utility is explained briefly with examples. The booklet also summarizes all features of Microsoft's Basic-80 (including the compiler) and Compiler System's CBasic. Error codes for CBasic-1, CBasic-2, and Basic-E are also summarized in a single alphabetical list. Examples and definitions explain the workings of DESPOOL, MAC and TEX utilities offered by Digital Research.

The introductory price of $3.75 includes postage and handling. For more information contact Rainbow Associates, P.O. Box 35, Glastonbury CT 06025. *Circle No. 135*

WHAT'S COMING UP

Super Isolator

Severe AC power line spikes, surges, noises and hash are prevalent in many microprocessor installations. Operators are plagued with unexplained crashes, memory loss and program glitches. Disks, printer and processor often interact, aggravating the problem.

Electronic Specialists' recently announced Super Isolator is designed to curb these severe electrical problems, the company said. Incorporating heavy duty surge/spike suppressors, the Super Isolator features 3 individually dual-Pi filtered 3-prong AC sockets. Equipment interactions are eliminated and disruptive/damaging line spikes and hash are controlled. The Super Isolator can accommodate an 1875 watt load.

Severe power line spike/surge and hash control is combined with interaction-free operation for $85.95 (Model ISO-3). For more information contact Electronic Specialists, Inc., 171 South Main St., Natick, MA 01760; (617) 655-1532 *Circle No. 124*

CLASSIFIEDS

ADVERTISER'S INDEX

Circle Number		Page
21	Adventure International	74
51	AEM Computer Services	109
43	American Square Computers	108
1	Atari Personal Computers	C-4
50	Audio Video Systems	109
23	Barclay Bridge Supplies	90
53	Berliner Computer	109
4	Bottom Shelf	3
57	CalData Systems	111
47	Joseph J. Charles	109
26	Computer Information Exchange	99
37	Computer Resources, Disks, etc.	108
32	Computer Systems Design	80
9	CPU Shop	12
25	Creative Discount Software	98
24	C&S Electronics Mart	98
56	Cygnus Systems	111
38	Data Processing Consultants	108
44	Discount Data Forms	108
28	Edu-Ware	100
27	Electronic Specialists	99
45	Fuller Software	109
52	Golway Associates Consulting	109
17	Mark Gordon Computers	51
34	Hayden Book	106
10	Hayes Microcomputer Products	21
2	Heath	C-2
14	H&E Computronics	35
8	Information Unlimited Software	11
33	Institutional Computer Development	80
48	Interface Technology	109
36	M. G. Kelly & Assocs.	108
13	Level IV Products	33
30	Microcomputer Technology	101
41	Micro Design	108
11	Micro Management Systems	23
29	Micro Peripheral	100
39	Midwest Computer Services	108
3	Mountain Computer	1
5	Netronics R&D	4
6	NRI Schools, Electronics Div	5
35	Omnico	107
42	Omnitek Systems	108
	Personal Computing	4,69,104,105,110
19	Programma International	55
16	Racet Computes	45
7	RCA Electro Optics	9
49	Jeanne St. Auber	109
	Sinclair Research Ltd.	53
31	Small Business Systems Group	102
22	Sun-Research	103
54	TYC Software	110
58	Max Ule & Co	112
46	Universal Interface	109
59	Vista	C-3
40	Wallace Computer	108
15	Xymec	37

SALES OFFICES

- **Northeast:** Arlene Joseph, Benwill Publishing Corp., c/o Morgan-Grampian, Inc., 2 Park Ave., New York, NY 10016; (212) 340-9780.
- **California:** Michael Reynolds, 924 Westwood Blvd., Los Angeles, CA 90024; (213) 478-3017.
- **Japan:** K. Yanigihara, International Business Corp., 10-10 Shinjuku 3-chome, Shinjuku, Tokyo, 160 Japan: (03) 350-0272.

When you pick a Daisy...
Pick Vista's V300 Printer.

$1895.

There are Daisies!. . . And, There are Daisies!. . . But Vista has a Peach!

The Vista V300 is exactly that, a "peach" of a daisy wheel printer both from the standpoint of price and performance.

Think of it, a printer at nearly half the price (when compared to models even remotely competitive in quality) combined with the ultimate in reliability, print quality, and flexibility.

Typical Comments: "Superb print quality!", "Highly reliable.", "Definitely letter quality. . . I can't believe the price tag.", "Best use I've seen yet of LSI Technology."

But judge for yourself — look at the V300 features and keep in mind this is a letter quality printer at dot matrix prices.

- **Tractor option available**
- **Print Speed — 25 CPS (Optional 45 CPS for $2,195)**
- **Print Wheel** — Industry standard 96-character Daisy Wheel (including the extended-life dual plastic wheels)
- **Service** — Prompt maintenance/service agreements available nationwide
- **Interface** — Industry standard parallel (RS232-C optional)
- **Printable Columns** — 136
- **Warranty — 90 days parts and labor, one year parts only**
- Proportional, bi-directional printing • Programmable VFU
- Extensive self-test functions • Hardware and software compatible

Vista does it again! Quality, Price and Performance with a peach of a daisy wheel printer.

AND, Vista Has a Complete V100 Word Processing System for Only $4995!

The Vista V100 is a complete word processing system that includes:

- Exidy Sorcerer Computer, 48K
- V200E20 Disc Drive System, Double Density
- Sanyo Data Display Monitor
- Vista V300 Printer Full Character Daisy Wheel
- Wordstar, CPM 1.4 (Includes E Basic)
- Can also be used for Data Processing

Vista

IMMEDIATE DELIVERY
For Further Information
Call Toll Free (800) 854-8017

The Vista Computer Company 1401 Borchard Street • Santa Ana, California 92705 • 714/953-0523

CIRCLE 59

ATARI®: PERSONAL COMPUTER SYSTEMS THAT GROW WITH YOU

Start with a better computer. Atari computers have built-in capabilities you can't even add onto many other personal computers. Three programming formats (ROM cartridge, disk and cassette). A 57 key upper/lower case ASCII keyboard with 29 keystroke graphics symbols, 128 colors and hues. Four separate sound channels and a built-in speaker. Four controller ports. A built-in RF modulator and FCC approval for connection to any TV. Plus, nationwide Atari Authorized Service Centers. And more.

Add memory. The ATARI 800™ is supplied with 16K of memory. You can expand up to a full 48K of RAM with 8K or 16K Memory Modules™ you install yourself. In less than a minute. The ATARI 400's™ 8K of RAM may be expanded to 16K at Authorized Service Centers. Both may be expanded to 26K of ROM with slip-in ROM cartridge programs.

Add peripherals. The ATARI 410™ audio-digital program recorder. Single or dual density* disk drives. The ATARI 800 individually addresses up to four drives. Add the ATARI 850™ RS232 Interface Module. Add high speed 40 or 80-column printers. Add an acoustic modem for remote data access. Add a light pen.* And there are more Atari peripherals on the way.

Add programs. Choose among dozens of programs in Atari's rapidly expanding software library. Programs categories include:

- **Personal Finance & Record Keeping**
- **Personal Interest & Development**
- **Professional Applications**
- **Education**
- **Information & Communication**
- **Entertainment**
- **Programming Languages**
- **Small Business Accounting**

Add it up. With Atari, you start with more. And you can build to more. Because Atari offers you personal computer systems that grow with you. Ask your Atari retailer to give you a full demonstration of Atari computers, peripherals and programs. Complete systems. Because when other people were thinking hardware and software, Atari was thinking systems.

*Available Fall, 1980

ATARI®
PERSONAL COMPUTERS
1265 Borregas Avenue, Sunnyvale, CA 94086

Call toll-free (800) 538-8547 (Except Alaska and Hawaii)
(In California: (800) 672-1404) for the name of your nearest Atari retailer.

CIRCLE 1